

# INFODIDACT'2021

## 14. Informatika Szakmódszertani Konferencia



Előadaskötet

2021

Szerkesztette: Dr. Szlávi Péter, Dr. Zsakó László  
Megjelenés: 2022. január

© 2022 Webdidaktika Alapítvány

**ISBN** 978-615-80608-5-1

## Tartalom

### Három intézmény – három megoldás Távoktatás pandémia idején

Bakonyi Viktória, Illés Zoltán, Heizler Adina, Pšenáková Ildikó ..... 5

### Digitális kultúra – Prezentáció – Tudástranszfer

Csernoch Mária, Csernoch Júlia ..... 17

### Témák és ötletek az információkezelés tanítására

Holló Csaba ..... 35

### Digitális eszközökkel támogatott verstani kutatások<sup>49</sup>

Gortva Tamás, Kelemen Péter, Kelemen András ..... 49

### Offline szituációs játék informatikai rendszerek oktatásához

Korom Szilárd, Illés Zoltán ..... 55

### Elemi algoritmusok – prefix összegzés

Menyhárt László Gábor, Zsakó László ..... 69

### Online programozás szakkörök – a tehetségnevelés jövője?

Nikházy László ..... 79

### A középiskolai tanulmányok hozadéka a számítógépes gondolkodás perspektívájából

Osztian Pálma Rozália, Kátai Zoltán, Osztian Erika ..... 91

### PowerPoint prezentációk helyessége különböző korosztályokban

Papp Gréta, Csernoch Mária ..... 113

### Interdiszciplináris Informatikai Tehetségek

Sarmasági Pál ..... 129

### Programozási alapfogalmak sorrendje a módszeres és funkcionális programozásban

Szabó Zsanett, Visnovitz Márton ..... 139



# Három intézmény – három megoldás Távoktatás pandémia idején

Bakonyi Viktória<sup>1</sup>, Illés Zoltán<sup>2</sup>, Heizler Adina<sup>3</sup>, Pšenáková Ildikó<sup>4</sup>

{<sup>1</sup>bakonyi, <sup>2</sup>illes}@inf.elte.hu, <sup>3</sup>adina.heizler@ohebs.hu,  
<sup>4</sup>ildiko.psenakova@truni.sk  
ELTE IK, Orchidea Iskola, Trnavská Univerzita v Trnave

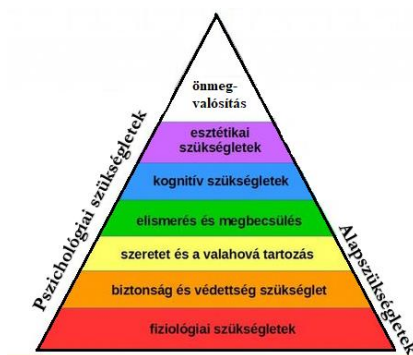
**Absztrakt.** A XXI. században aligha lehet eléggé hangsúlyozni az oktatás fontosságát. A nyelv- ismeret, az informatika ma már az alpműveltség része, nélkülük sikeres életpálya szinte elkép- zelhetetlen, bármilyen jövőbeli szakmát is választ magának a tanuló. Ezt ismerik fel egyre töb- ben és emiatt is válik ez a két terület egyre hangsúlyosabbá. Ma egyre gyakoribb az is, hogy a diák nem a hazájában, hanem külföldön, esetleg nem is az anyanyelvén tanul, jellemzően angol- ul. Meghatározhatja a gyermek, a fiatal egész további életét, ha belépve az oktatási intézmény kapuján valami miatt nem tudja felvenni a lépést, és lemarad tanulmányaiban. Ezt az amúgy sem könnyű időszakot nehezítheti tovább, ha a vészhelyzet miatt éppen távolléti oktatásban kell részt vennie. Eukleidész „*a geometriáboz nincs királyi út*” parafrázisa, hogy a beilleszkedés bi- zony komoly erőfeszítésbe kerül mindenki részéről. Az integrálódás nehézségeiben az okta- tóknak segítenie kell a hallgatóit minden rendelkezésére álló eszközzel, persze az se haszontal- an, ha a diákok, hallgatók is átérzik a helyzet komolyságát. Emiatt a tanárképzésben is gondot kell fordítani erre a területre. Természetesen a diákok részéről is szükséges az átlagosnál na- gyobb erőfeszítés és tenni akarás, amelynek hiánya az eredményt is befolyásolja. Cikkünkben a saját gyakorlatunkon keresztül mutatjuk be ezt a kihívást.

**Kulcsszavak:** oktatás, valós idejűség, virtuális osztályterem, online oktatás, hibrid oktatás, for- dított osztályterem

## 1. Bevezető

A Covid-19 által okozott pandémia 2020-ban gyökeresen megváltoztatta az egész világon az okta- tást, hiszen mindenkinek, a legkisebbeknek is távoktatásra kellett áttérnie. Egyfelől sok helyen esz- közhiánnyal, technológia nehézségekkel is szembesülnie kellett a résztvevőknek, másfelől az oktatás személytelenebbé vált, az emberi kapcsolatok háttérbe szorultak, miközben a diák-tanár pozitív viszony és a kortárs közösség erős motivációs hatással bír. Az e-learning megjelenésével már a 2000- es években kutatások bizonyították, hogy a motiváció csökken online környezetben. A magyarázat nem egyszerű, sokféle, komplex hatás játszik közre, de például az izoláció hatása is tetten érhető. Később, az elterjedő és népszerű MOOC kurzusok magas kiesési arányát is sokan ezzel indokolták [1]. Talán éppen ez az, amiért mindhárom, a cikkben tárgyalt intézmény ragaszkodott az online szinkron órák megtartásához.

A fentiek miatt nyugodtan mondhatjuk, hogy az új iskolát távoktatásban kezdő diákok szinte légüres térbe kerültek, hiszen nem ismerték sem a tanáraikat, sem pedig a diáktársaikat. Nézzük a két végletet, az iskolába éppen belépő elsősöket, akik igénylik a folyamatos segítséget és az egyetemre bekerülő diákokat, akik eddig a középiskolában folyamatos kontroll alatt voltak, de egyetemistaként már maguknak kell felelősséget vállalni a tanulásukért (is). Ez sokaknak a távoktatás előtt is gondot jelentett, hiszen minden ember igényli az elfogadást, a valahová tartozás élményét lásd 1. ábra.



1. ábra: Maslow piramis

A tanárképzésben ebből a szempontból érdekes kettősséggel rendelkezik, hiszen egyfelől diákok, másfelől leendő szakmájuk az oktatás. Számukra igen fontos a tanítási gyakorlat, hiszen az addigi elméleti ismereteket ott tanulják meg alkalmazni a gyakorlatban is. Ez alatt tanulják meg, hogy is lehet és kell egyszerre figyelni az egész osztály haladására és közben személyes figyelemben is részesíteni a diákokat. Érzékeny probléma, ha éppen ebben az időszakban történik meg az iskolák bezárása, így hiányossá válhat egy komplex folyamat.

Ugyanakkor a már gyakorló tanárok, tanítók helyzete sem volt könnyű, főleg az elsősök esetében. Igaz, az általános iskola elsősei nem mindjárt távolléti oktatással indítottak 2020 szeptemberében – csak később és rövidebb időre kellett átállniuk erre a módszerre, de így is megterhelő volt számukra és családjaiknak ez az időszak. Gondoljunk csak bele, az eszközhasználat ebben a korban nincs, vagy nagyon hiányos – még az írás vagy olvasás biztonsága is kérdéses. Ráadásul és az iskolák egy része még csak most tanulja a hatékony IKT használatot [2][3]. A kutatók ma külön figyelmet fordítanak erre a problémára, feltárva mind a szülők, mind a tanárok véleményét. Kompetencia méréseket is végeztek a digitális oktatás eredményességéről, koncentrálna a legérzékenyebb korosztályra, az iskolakezdőkre. Gyakran az oktatók, a szülők technikai tudása és a még kiforratlan módszerek is hátráltatták a tanítást. Többen észlelték a saját gyakorlatukban is, hogy az otthoni tanulás alatt a kisdíjakok motiválatlanabbnak bizonyultak. A sikerhez a *tanárok és szülők szorosabb együttműködése* és minden résztvevőtől *nagyobb munka befektetése* szükséges. [4]

Végezetül hagy szóljunk az informatikus hallgatóinkról. 2020 őszén felmérést végeztünk, hogy elsős (egyetemista) diákjainknak mi okozta a legnagyobb nehézséget. Az egyetemisták persze már nem szorulnak annyi segítségre. Az informatikus hallgatóknak pedig általában rendelkezésre állnak a szükséges eszközök és a távtanuláshoz szükséges informatikai ismeretek. Az akkori golyáink viszont egy teljes évig be sem léphettek az egyetem kapuján és ráadásul az azt megelőző érettségi, felvételi időszakuk is eltért a hagyományostól. A válaszok között dominánsan szerepelt a személytelenség érzése.

1. táblázat: Távoktatási nehézségek ELTE IK [5]

	Csökkenett figyelem	Személytelenség	Koncentrációs probléma
Magyar elsősök	13,19%	43,96%	24,18%
Angol elsősök	15,79%	14,04%	22,81%

Az idei, 2021/2022-es évben pedig olyan elsősök kezdik egyetemi tanulmányaikat, akik a középiskolás éveik végének jó részét távolléti oktatásban töltötték és most pedig a szintén újszerű hibrid-oktatási módszerrel találkoztak. Szlovákiában a helyzet még érdekesebb, mert a középiskolákon elmaradt az érettségi és az érettségi bizonyítvány jegye a négy évi átlag lett. A távoktatás, a virtuális

osztályterem alkalmazások, az online számonkérések pedig gyökeresen megváltoztatták a tanulási szokásokat.

## 2. Tanárképzés, Trnava – fordított osztályterem módszer

A Trnavai Egyetemen (<https://www.truni.sk/en>) többek között tanárképzés is folyik. A pedagógiai gyakorlat teljesítése komoly problémát jelentett leendő pedagógusaink számára. Az iskolák be voltak zárva, a diákoknak nem volt lehetőségük lefolytatni a megszokott tanítási gyakorlatot és személyesen oktatni. De ha a másik oldalát nézzük a helyzetnek, akkor pozitívumnak számíthat, hogy kipróbálhatták az online tanítást nem csak, mint diák, hanem mint tanár is. Mi lehetne hasznosabb, mint saját magukon érezni egy speciális helyzetet, amelyet későbbi pályájukon tetszőleges távköztetési környezetben alkalmazhatnak? A megismert digitális eszközök mit sem érnek a tanár nélkül, aki élettel tudja megtölteni az online anyagokat.

*"A tanár elengedhetetlen eleme az elektronikus eszközök használatával történő tanításnak. A tanár az, aki jól ismeri tanítványait, és a legjobban fel tudja mérni, hogy a tanítási folyamat mely szakaszában, milyen formában, de azt is, hogy milyen tartalommal célszerű az elektronikus programokat használni."* [14]

### Virtuális osztályterem, személyes figyelem

Nemcsak a diákjaink számára jelentett nehézséget a tanítási gyakorlat lebonyolítása, hanem nekünk, egyetemi oktatóknak is döntést kellett hozni arról, hogyan segíthetjük legjobban tanítványainkat a tanulásban, a tanításban. Ezt több tantárgyban *fordított osztályterem* (flipped classroom) módszerrel oldottuk meg, vagyis az előadásokra videofelvételeket készítettünk és ezt minimum két nappal az előadás órája előtt feltöltöttük az MS Teams-ben kialakított tanulmányi csoport fájlok közé. A diákoknak így lehetőségük volt felkészülni a jövőbeli órára. Maguk az órák online folytak az MS Teams-ben, és leginkább konzultáció formájában tartottuk, vagyis kérdez-felelek stílusban.

### Közös munka

Mivel tapasztalataink elsősorban az informatikai tantárgyak oktatásához kapcsolódnak, a diákoknak gyakran kellett feladatokat megoldani (program, adatbázis, interaktív tananyag készítés), ahol nagy hasznát vettük a képernyő megosztásnak. Voltak esetek, amikor a diák kérdésére a tanár megkérte, hogy ossza meg a képernyőjét, és navigálta a diákok hogyan oldja meg a problémát. Természetesen ezt a többi diák is látta, és volt olyan eset, hogy a 152 résztvevő diák közül 73 jelezte, hogy sikerült megoldaniuk ugyanazt a problémát.

Ez a tapasztalatunk jelzi, hogy vannak lehetőségek az online oktatásban is, amelyek segítenek áthidalni a problémákat, de sajnos, ennek megvalósítása nem minden tantárgyban és minden korosztály oktatásában kézenfekvő. Nézzünk egy olyan esetet, amelyek a legnehezebbek közé tartoznak: írni, olvasni tanítás első osztályban.

## 3. Közoktatás, Budapest – vegyes tanulás módszer

Az Orchidea (<https://althajdu.ohebs.hu>) Általános Iskolai (Hajdú utca) részébe minden évben 42-46 tanulót vesznek fel, melynek körülbelül negyed része nem magyar anyanyelvű. Ezek a gyerekek a magyar anyanyelvű tanulókkal kerülnek egy osztályba és várhatóan Magyarországon kívánnak majd érettségizni (magyarul). Azon gyerekek, akik intézményünk óvodájába jártak, de nem magyar anyanyelvűek, könnyedén tanulják meg a nyelvet, mire első osztályba lépnek. Minden elsőst felmérnek a kollégáink, hogy milyen nyelvi szinten állnak, szükségük van-e MMD (magyar, mint idegen nyelv) órákra. Ezekre az órákra a pedagógus azonos nyelvi szinten lévő tanulókat válogat össze.

Iskolánkban már első osztálytól vannak olyan tantárgyak, amelyeket angol nyelven tartunk. Ilyen a vizuális kultúra, technika és tervezés, informatika (jövő évtől digitális kultúra), testnevelés, és ter-

mészetesen az angol nyelv. A többi tantárgyat magyarul oktatjuk, mint az etika, ének-zene, matematika, magyar nyelv és irodalom (olvasás, írás, nyelvtan, fogalmazás), környezetismeret. Egyes tárgyakat később angol nyelven is tanulják a gyerekek heti egy órában, mint a természetismeret és matematika.

### **Virtuális osztályterem**

Az online oktatás alatt intézményünk a *Google Classroom* szolgáltatását vette igénybe a tananyagok megosztására, valamint feladatok beküldésére, ellenőrzésére. Ezen felül a *Zoom*-on keresztül tartottuk meg online az óráinkat.

A diákjaink és szüleik számára próbáltuk a lehető legkönnyebben kezelhetővé tenni ezeket a heteket/hónapokat, így minden osztály létrehozott egy online elérhető órarendet, melyben jelöltük, hogy mely órák az online megtartandóak, illetve melyek, amelyek offline valósulnak meg. Az online megtartandó órákra az adott pedagógus *Zoom* linkjét helyeztük el, így a diákoknak csak kattintaniuk kellett a megfelelő tanórára. Figyelembe vettük, hogy az elsős diákokat ne terheljük túl egyik napon se. Nagy változás számukra az, hogy otthonról tanulnak, így abban maradtunk, hogy a magyar nyelv (olvasás, írás), matematika, környezetismeret, angol meghatározott heti számban, de online folyjon, míg a többi tárgy offline, feladatküldés segítségével.

### **Személyes figyelem, interakció**

Az online órák alatt, kolléganőmmel felosztottuk az osztályt két csoportra, hogy minél több személyes interakcióra legyen lehetőség. Amíg az osztály fele matematikára csatlakozott be kolléganőmhöz, addig a másik fele hozzám kapcsolódott be olvasásra vagy írásra. Ezzel a módszerrel megfelelően tudtunk haladni miközben a gyerekek szinte folyamatosan hozzá tudtak szólni vagy kérdezni. Ez számunkra kétszer annyi megtartott órát jelentett az online oktatás alatt.

Igyekezünk az offline megtartott órákat is személyesebbé varázsolni a gyerekek számára (*vegyes tanulás = blended learning*). Rövid videókat vettünk fel egy *Google* bővítmény segítségével, a *Loom*-mal. Ezeknek a kis, rövid videóknak a linkjét osztottuk meg a *Classroom*-ban. Rögtön az első ilyen videó után jött szülői visszajelzés, hogy nagyon tetszett a tanulónak ez a típusú offline óra, mert hallhatta a diák a hangomat és együtt dolgoztunk végig.

### **Technikai, módszertani trükkök**

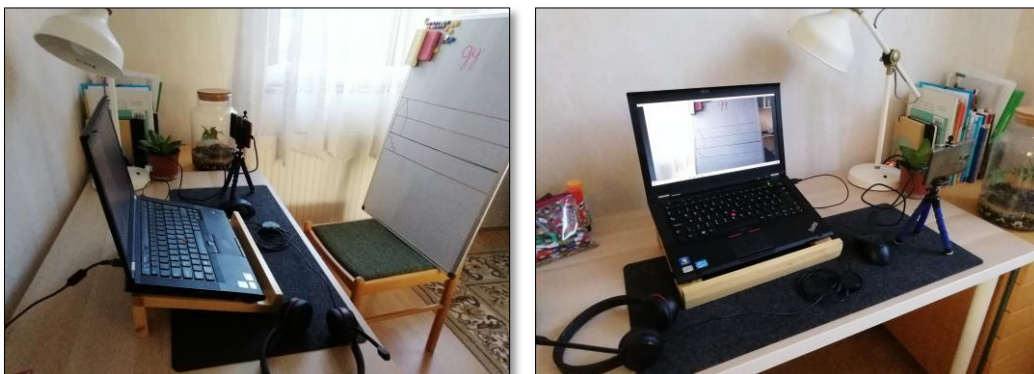
Az online-offline órák folyamán alkalmaztam egy külső kamerát is. Erre azért volt szükség, mert a személyes oktatás alatt még nem tanultunk meg minden betűt leírni a gyerekekkel, így szükség volt arra, hogy a gyerekek ne csak az arcomat, hanem a kezemet is lássák. Minden alkalommal előkészítettem egy kicsi fehér táblára az órán használandó vonalakat (íráshoz), és egy alkalmazás segítségével (*DroidCam*) a telefonom kameráját csatlakoztattam a *Zoom*-ba lsd 2. ábra. Ennek köszönhetően a gyerekek látták a táblát, valamint engem, és együtt tanultuk meg a betűk alakítását pont úgy, mintha az iskolában lennénk.

A tanulók nagyon ügyesen alkalmazkodtak az online oktatáshoz. Az első napokban még segítségre szorultak a *Zoom*-hoz való kapcsolódásban, de a hét közepére megtanulták, hogy mikor hova kell kattintani. A szülők segítségére az egész online oktatás folyamán szükségünk volt, hiszen a feladatokat a gyerekek nem tudták önállóan feltölteni a megfelelő felületre. Ezen a felületen, mi minden beadott feladatot értékeltünk, hiszen nagyon fontos a visszajelzés online oktatás folyamán is.

Sok-sok játékot próbáltunk becsempészni a gyerekeknek az online oktatás folyamán is, figyelembe véve az életkori sajátosságukat.

Azon tanulók, akik nem magyar anyanyelvűek és járnak MID órára, online módon folytathatták a tanulást, így nem állt le az ilyen típusú munka sem.





2. ábra: Írás tanítása

### Visszacsatolás

A felénk érkezett szülői visszajelzések alapján, nagyon elégedettek voltak az online oktatással. Az első hét után érdeklődtünk kolléganőmmel a szülők véleményéről. Általánosságban az mondható el, hogy azon szülők küldtek véleményt, akik szeretik megosztani gondolataikat személyes oktatás folyamán is. Kérdéseink között szerepelt az online/offline órák elosztásának aránya, feladatok mennyisége és a feladatok feltöltésére a megadott idő elég-e. A legtöbb kérdést a feladatok feltöltésével kapcsolatban kaptuk. Igyekeztünk minden offline óra után elegendő időt hagyni a feladatok feltöltésére, de a szülői visszajelzések alapján erre több időt adtunk. Az a kérdés merült fel még a szülőkben, hogy az offline órákat akkor kell-e elvégezni, amikor az órarendben van az ideje, vagy lehet később is. Terveink szerint az órarendi időt szántuk erre, de észlelnünk kellett, hogy szükségük van a gyerekeknek az órák közötti több játékra és szabadban való tevékenységre, így ezeknek az óráknak a beosztását a szülőkre bíztuk.

2. táblázat: Szülői visszajelzések

	Darab
Megkérdezett szülők	23
Pozitív visszajelzést adtak	6
Negatív visszajelzést adtak	-
Több időt kért a feladatok feltöltésére	4

A gyerekek az online oktatás végére elfáradtak és nagyon hiányzott nekik az iskola, valamint a közösség. Nagy mosollyal tértek vissza az iskola falai közé.

Végezetül nézzük az informatikus képzésünket, ahol az eszközhasználat kevésbé okozhat problémát.

## 4. Programtervező informatikus képzés, Budapest – szinkron online módszer

Az ELTE Informatikai Kar (<https://www.inf.elte.hu/>) évente 6-700 magyar hallgatót vesz fel, de 100 körüli az angol nyelvű képzésre felvettük száma is. A távolléti oktatás időszaka a magyar hallgatók számára is megterhelő volt, de az angol képzésben részt vevőknek már az egyetem megkezdését

is jelentősen megnehezítették a vészhelyzettel együtt járó utazási korlátozások és a karantén kötelezettség.

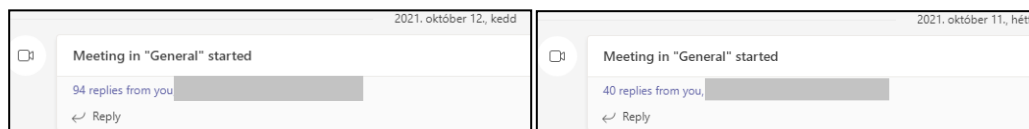
### Virtuális osztályterem

Az *online szinkron oktatás* megvalósítása alapvetően a *Microsoft Teams* segítségével folyt – illetve a *hibrid oktatásban* az éppen távollévőknek ezen keresztül sztreameljük az órákat, amelyekről felvételek is készülnek. A tanulást segítő, az előadás prezentációk, gyakorló feladatsorok és a video felvételek a teljes félév során végig hozzáférhetőek. A karon a *Canvas* rendszert, mint LCMS-t használjuk az oktatási tartalmak, a félévközi eredmények, hiányzások adminisztrálására, a *Neptun* pedig továbbra is az egyetem LMS rendszere. A késvé csatlakozó külföldi diákok számára elég nagy kihívást jelent, hogy önállóan, családi és baráti háttér nélkül kell boldogulniuk egy idegen országban. Kutatás bizonyítja, hogy a *kapcsolati beágyazódás* elősegíti a diploma megszerzéséhez szükséges kitarító munkát. Ez a beágyazódás egyfelől a hallgatótársakkal való kapcsolatot, másfelől az intézménybe és az oktatókba vetett bizalmat is jelzi. [6][7]

### Személyes figyelem, interakció

Az általános beilleszkedésüket segíti a mentor rendszer és a tanulás-módszertani kurzus vagy az életvezetési tanácsokkal is szolgáló diák tanácsadó központ. A *Stipendium Hungaricum* ösztöndíjat elnyert tanulóknak kötelező magyar órájuk van 6 kredit értékben, hetente 2 x 2 órában, hogy megismerhessék annak az országnak a nyelvét és kultúráját, ahol egyetemi éveiket töltik. A saját óráimhoz konzultációs lehetőséget adtam minden diákomnak előre megadott időintervallumokban, hetente többször is.

2020 tavaszi félévében volt alkalmunk tapasztalni, hogy a hallgatók nem szívesen szólnak bele a mikrofonba, inkább a szöveges üzeneteket preferálják. [8] Ez a tapasztalat vezetett bennünket arra, hogy az óra közben igen sokszor chaten keresztül, „like”-kokkal is kérjünk visszajelzést a hallgatók aktuális gondjairól, ami kedvelt, a hétköznapokban is bevált kommunikációjuk. Különösen az angol képzésben résztvevők élnek ezzel a lehetőséggel igen gyakran. Előfordul, hogy egy-egy óra alatt 90-100 üzenetet is küldenek, lásd 4. ábra.



4. ábra: Angol és egy magyar nyelvű óra chat használata

A magyar hallgatók ennél sokkal kevésbé használják ezt a lehetőséget. A képernyőjük megosztása kevésbé gyakori, pedig sokat segíthet egy-egy hiba megtalálásánál.

Az előadásokon a chat üzenetek helyett egy saját készítésű CRS-t (Classroom Response System) alkalmazást, az E-Lectiiont használjuk a nagylétszámú hallgatóság aktivizálásához [9][10]. A rendszer valós-idejű, kétirányú webes alkalmazás, amely alkalmas az azonnali visszajelzések begyűjtésére és későbbi adatfeldolgozásra is. Használata kényelmes bármilyen eszközön, mobil telefonon, tableten vagy lapon, így a normál oktatás során is segítette a munkánkat az előadóterekben. [11][12][13]

### Visszacsatolás

Az egyetemisták felnőttek, önállóak, de visszajelzésekre szükségük van, hogy felmérhessék hogyan is állnak az aktuális tudásukkal. A személyes kapcsolatok hiányán túl, nem ismerték az online vizsgáztatási felületet sem és komolyan aggódtak életük első egyetemi zárthelyijének nehézsége miatt. Mind a magyar, mind pedig az angol hallgatók legelső nagyobb zárthelyije Számítógépes rendszerek/Computer System tantárgyból van az első félévben. 2020 őszén, az első távoktatással indult

félévben lehetett érzékelni, hogy a diákok bizonytalanabbak, mint korábban – sem a tanáraikat, sem egymást nem ismerték.

Azért, hogy oldjuk a feszültségüket, minden kiválasztott csoportunk számára publikáltunk a *Canvas* rendszerben egy próba dolgozatot, hogy kipróbálhassanak mindent előre és megnyugodjanak. A dolgozat file feltöltős kvízként készült, ahol az egyes feladatokat egy-egy feladatcsoportból véletlenszerűen sorsolja a rendszer, így biztosítva, hogy minden feladatsor lehetőleg különböző legyen.

Ugyanezeket az idei hibrid évben is megismételtük – bár minden második héten bejöhettek a kampuszra, de a félév negyedik hetében még elég ismeretlen lehet minden a számukra.

Három különböző típusú csoportunk volt/van:

- Válogatottan jók (450 pont feletti, de nem a legjobbak, nem a verseny győztesek csoportja)
- Normál (450 pont alatti hallgatók)
- Angol képzésben (Vegyes előképzettséggel rendelkezők)

### Nincs királyi út

Ahogy a bevezetőben már írtuk, a beilleszkedéshez, egy új életszakasz az egyetemi élet megkezdése erőfeszítést igényel a hallgatóság részéről is, „nincs királyi út”. A sikerért meg kell dolgozni. A többség örült a próba zh lehetőségnek, de a legnagyobb meglepetésemre voltak, akik nem is foglalkoztak vele. (3. táblázat, 3. ábra) Ráadásul az idei 2021 szeptemberi félévében csökkent azok aránya, akik kipróbálták magukat az éles zárthelyi előtt.

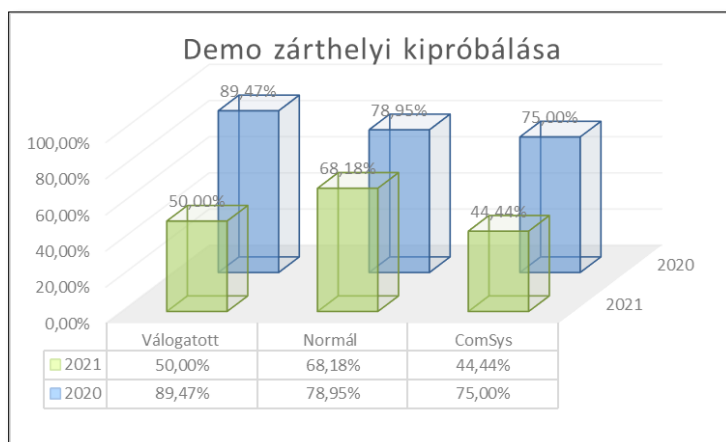
Többféle oka lehet ennek a csökkenésnek.

- Az angol hallgatók kezdése a szokásosnál egy héttel korábbra lett megadva, így kevesebb idejük volt a magyarországi életkezdésük ügyintézéséhez pl. TB kártya, bankkártya stb. Másfelől a hibrid oktatásban a jelenlét azért megvalósul, így személyes kapcsolatok is kialakulnak – egymástól is jobban kapnak információt, korábbi évek zárthelyijeit is közösen foroghatnak, így talán nincs akkora szükségük erre a lehetőségre.
- Úgy gondoljuk, hogy a megváltozott tanulási és értékelési szokások is megjelennek ezekben a tendenciákban. Ebben az évben az elsős tanulóink már másfél évet a távoktatás kereteiben töltöttek el – nem tartanak annyira ettől az oktatási formától.
- A válogatottan jók érezhetnek kellő önbizalmat, sokan tanulhattak korábbi iskolájukban ezekről a témákról, illetve sokan számítógéppel foglalták le magukat a kényeszerű bezártság idején.

A normál csoportban csökkent a legkevésbé a próba zh-t megírók száma – ők kevésbé lehetnek magabiztosak a felvételi pontszámuk alapján és jó részük az iskolában nem tanulhatott túl sok programozást. Az angol csoport, ahol 50% alá csökkent a kipróbálók aránya, véleményem szerint a fentebb már említett korábbi kezdés áldozatai.

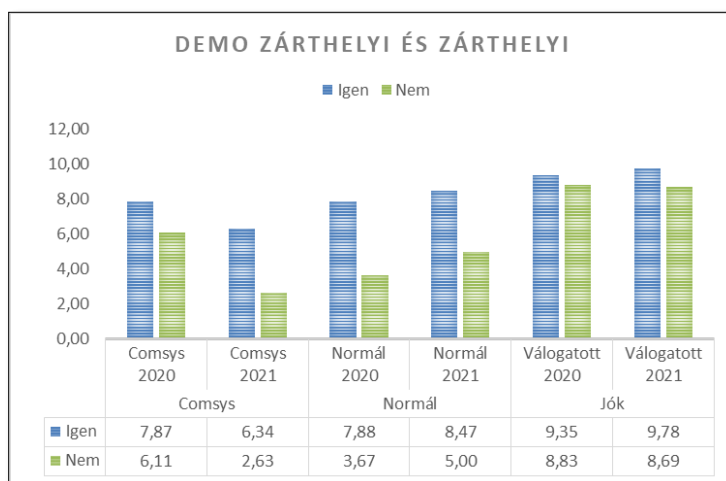
Csoport	Kipróbálta a demo zárthelyit	
	Igen	Nem
Válogatott 2020	17	2
Normál 2020	15	4
Angol 2020	15	5
Válogatott 2021	9	9
Normál 2021	15	7
Angol 2021	16	20

3. táblázat: Próba zárthelyi kipróbálása 2020-2021



**5. ábra:** Demo zárthelyik kipróbálása

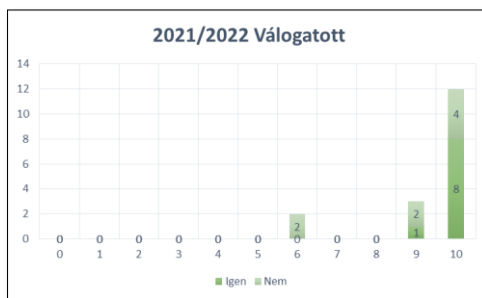
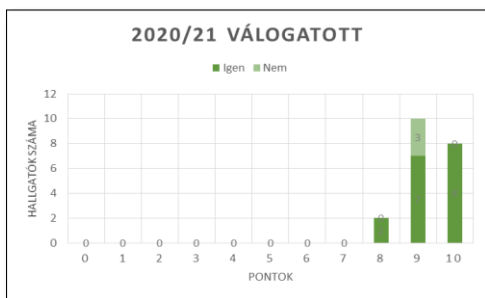
Nyilván adódik a kérdés, hogy van-e összefüggés a próba zárthelyi megnyitása, megnézése és az éles zárthelyin elért eredmény között. Véleményünk szerint ez egyértelműen a hallgató hozzáállását jelzi a tanuláshoz. Habár tudjuk, hogy léteznek olyan kiváló képességűek, akiknek nem kell semmilyen erőfeszítést tenni az átlagos tudás megszerzéséért, a többség nem így van vele.



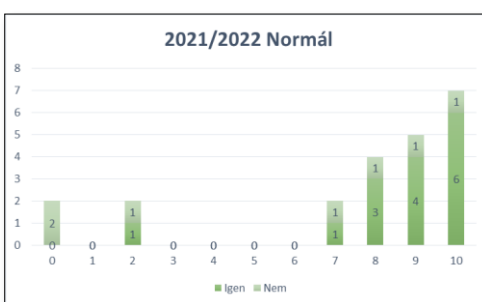
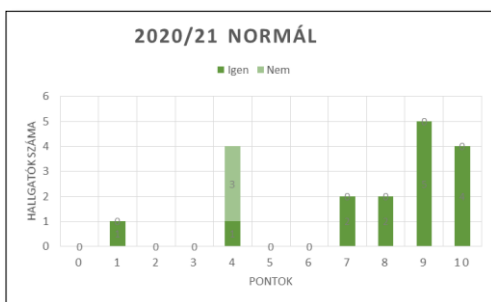
**6. ábra:** Demo zárthelyi kipróbálása és zárthelyi eredmény

Az első zárthelyin 10 pontot lehet elérni, 5 pontnál kevesebb esetén pedig elégtelen a teljesítmény. Az 6. ábrán azt láthatjuk, hogy minden csoport típus esetében azoknak a pontszáma a magasabb, akik időt szántak a próba dolgozatra.

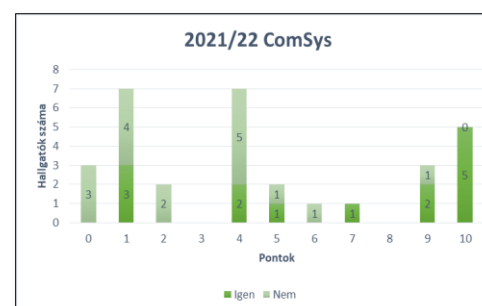
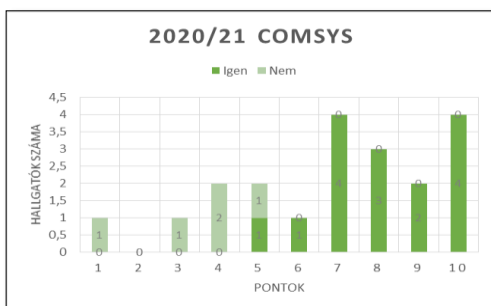
A legkisebb különbség az átlagpontok között a válogatottan jók esetében van – ők azok, akik könnyen veszik általában az akadályokat. Nagyobb erőfeszítés nélkül is tudják teljesíteni az első felmérést. A normál képességűeknél látszik, hogy milyen sokat számít a hozzáállás, a szorgalom, hiszen 1,5-2-szeres a pontokban mért eredmény különbsége. Az angol csoport eredménye 2020-ban a normál csoporttal volt azonos azoknál, akik kipróbálták a feladatokat, de a nem kipróbálók kisebb átlaga sem olyan mértékű, mint a magyar normál csoportnál.



7. ábra: Válogatottan jók eredményei



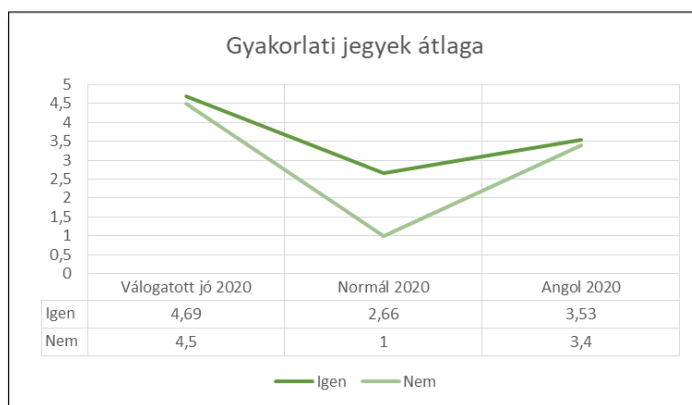
8. ábra: Normál hallgatók eredményei



9. ábra: Angol hallgatók eredményei

Az idei év drasztikus különbsége a korábbi évkezdes számlájára írható. Az 7.,8. és 9. ábrán az egyes csoportoknak megfelelő ponteloszlásokat látjuk. Teljesen egyértelműen azok kaptak magasabb pontokat, akik részt vettek a próba dolgozaton – talán ez a válogatottan jóknál tükröződik a legkevésbé. Sötétebb zölddel az „igen”, megírta a demót, világosabbal pedig az azt kihagyókat jelöltük.

Teljesen egyértelműen következik annak a vizsgálata, hogy a 3.-4. héten felkínált próba zh megírása vagy elhagyása a félév végi végső jegyben eltérést mutat-e. A kezdeti hozzáállás meghatározó-e az elért végső jegyben.



**10. ábra:** Gyakorlati jegyek a 2020-as őszi félévben

Jól látszik, hogy a végső jegyben már nem olyan szignifikáns a különbség 10. ábra, de azért minden esetben a „szorgalmasabb”, igyekvőbb, a próba zárthelyit is megírók teljesítenek jobban. A normál csoport 2020-as év végi eredményébe közrejátszik az, hogy többen feladták a jegyszerzést, így ebben a táblázatban elégtelennel zárult a félévük (a valóságban nem kaptak érdemjegyet).

Jól látszik, hogy milyen sokat számít az oktatás eredményességében a korábban kialakult pozitív attitűd a hozott tudásszinttel együtt, bár megfelelő tanítási módszerekkel a végeredmény kiegyensúlyozottabbá tehető, a kezdeti egyenlőtlenségek részben orvosolhatók.

## 5. Összefoglalás

Már Arisztotelész is megmondta, hogy az ember társas lény. A veszélyhelyzet alatt az online, digitális tanítás új módszereket, más megközelítéseket kívánt minden résztvevőtől. A három vizsgált oktatási intézményben online szinkron, vegyes / hibrid / fordított osztálytermi oktatás folyt. Az egyes módszereket az aktuális hallgatóság igényeihez, lehetőségeihez szabták, hogy a lehető legjobb eredményt ériék el. A bemutatott oktatási gyakorlatok azt igazolják, hogy a tanároknak jóval több energiát kellett befektetni ahhoz, hogy ki tudjanak alakítani egy támogató, a személyes figyelmet is alkalmazó környezetet, és bevonjanak minél több diákot a munkába. Az interakció, a folyamatos visszacsatolás eredményezheti csak azt, hogy megmaradjon a motiváció, erős maradjon a tanulni akarás mindenkiben. Ehhez természetesen megfelelő eszközökkel, technológiai ismeretekkel és ennek megfelelő módszertannal is rendelkeznie kell az oktatóknak és a diákság megfelelő hozzáállása is szükséges.

## Köszönet

EFOP-3.6.3-VEKOP-16-2017-00001: Tehetség gondozás és kutatói utánpótlás fejlesztése autonóm járműirányítási technológiák területén – A projekt a Magyar Állam és az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg

KEGA 012TTU-4/2021 “Integrácia využívania dištančných výučbových procesov a tvorby elektronických učebných materiálov do edukácie budúcich pedagógov (A távoktatási folyamatok alkalmazásának és az elektronikus tananyagok készítésének integrálása a leendő pedagógusok oktatásába) szlovákiai projekt támogatásának.

## Irodalom

1. Maggie Katherine Hartnett: *The Importance of Motivation in Online Learning*. [https://www.researchgate.net/publication/315033704\\_The\\_Importance\\_of\\_Motivation\\_in\\_Online\\_Learning](https://www.researchgate.net/publication/315033704_The_Importance_of_Motivation_in_Online_Learning) (utoljára megtekintve: 2021.10.31.). DOI: 10.1007/978-981-10-0700-2\_2
2. Ildikó Pšenáková: *Interaktív tesztek a tudásszint mérésére*. In: OXIPO. – ISSN 2676-8771 (online), č. 1 (2019), pp. 85–93.
3. Ildikó Pšenáková, Tomáš Godiš, Milan Štrbo: *Az interaktivitás szerepei az idegen nyelv oktatásában*. In: InfoDidact 2019 (Budapest: Webdidaktika Alapítvány, 2020.) – ISBN 978–615–80608–3–7 (online), pp. 245–252. <http://konferenciak.inf.elte.hu/infodidact/InfoDidact19/Manuscripts/PIGTSM.pdf> (utoljára megtekintve: 2021.10.31.)
4. Józsa Krisztián, Pasztendorf Gabriella: *Az olvasástanítás eredményessége az iskola kezdő szakaszában a Covid19 időszaka alatt: a pedagógusok és a szülők megítélés*. In: Gyermeknevelés Tudományos Folyóirat 9. évf., 2. szám 131–144. (2021) DOI: 10.31074/gyntf.2021.2.131.144
5. V. Bakonyi, Z. Illes, and C. Verma: *Virtual classrooms and real-time interactivity*. In 2021 2nd International Conference on Intelligent Engineering and Management (ICIEEM), 2021, pp. 69–74.
6. Pusztai Gabriella, Győri Krisztina: *Learning Alone – a kapcsolati beágyazódás vizsgálata a távolléti oktatás alatt*. In: Educatio 30 (1), pp. 112–124 (2021) DOI: 10.1556/2063.30.2021.1.9
7. R. Takács, A. Oláh, Z. Horváth: *Lemorzsolódás vizsgálata a Covid19-járvány következtében távolléti oktatásra váltott intézményben*. In: Online tanácsadás a felsőoktatásban a Covid19-járvány idején, 2020, pp. 63–74.
8. CNN, Amy Gahrn: *One-third of Americans prefer texts to voice calls*. <https://edition.cnn.com/2011/09/22/tech/mobile/americans-prefer-text-messages/index.html> (utoljára megtekintve: 2021.10.31.)
9. V. Bakonyi, Z. Illes, and C. Verma: *Towards the Real-Time analysis of Talks*. In: International Conference on Computation, Automation and Knowledge Management, 2020, pp. 322–327.
10. V. Bakonyi, Z. Illes, and C. Verma: *Analyzing the Students' Attitude Towards a Real-Time Classroom Response System*. In: 2020 International Conference on Intelligent Engineering and Management (ICIEEM), 2020, pp. 69–73.
11. V. Bakonyi, Z. Illes, and C. Verma: *Key element in online education to activate students with real-time tools*. In 2021 2nd International Conference on Computation, Automation and Knowledge Management (ICCAKM), 2021, pp. 326–331.
12. B. Viktoria, Z. Illés, and V. Chaman: *Real-time Education in Emergency Situation*. In :2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), 2021, pp. 1–6.
13. V. Bakonyi, D. Szabó, S. Korom, Z. Dr. Illés: *A távolléti oktatás tapasztalatai*. In: InfoDidact 2020 (Budapest: Webdidaktika Alapítvány, 2020.) – ISBN 978-615-80608-4-4 (online), pp. 5–11. <http://konferenciak.inf.elte.hu/infodidact/InfoDidact20/Manuscripts/BVSzDKSziZ.pdf> (utoljára megtekintve: 2021.10.31.)
14. Godiš, T.: *Integration moderner Sprachprogramme und Apps in den Fremdsprachunterricht*. In: Forlang [elektronický dokument]: periodický zborník vedeckých príspevkov a odborných článkov z medzinárodnej vedeckej konferencie konanej 23. - 24. júna 2021 Roč. 8, č. 1. Košice. Technická univerzita v Košiciach, 2021. – ISBN (elektronické) 978-80-553-3948-1. – ISSN 1338-5496, s. 99–110





# Digitális kultúra – Prezentáció – Tudástranszfer

Csernoch Mária<sup>1</sup>, Csernoch Júlia<sup>2</sup>

<sup>1</sup>csernoch.maria@inf.unideb.hu, <sup>2</sup>juliacsernoch@hotmail.com

<sup>1</sup>Debreceni Egyetem IK, <sup>2</sup>University of Neuchatel

**Absztrakt.** A *Digitális kultúra* tantárgy oktatása olyan keretek között kezdődött meg a 2020/2021-es tanévben, amelyek figyelmen kívül hagyják a tantárgyon belüli és a tantárgyak közötti tudástranszfer elemeket. Ennek következtében a már megjelent tankönyvek az egyes témaköröket egymástól függetlenül, a korábbi tantárgyi gyakorlatnak megfelelően, alacsony hatékonyságú felületi megközelítésekkel tárgyalják, az eszközökre koncentrálnak, szemben a számítógépes gondolkodás fejlesztésével. Jelen tanulmány a kerettantervi és tankönyvi elemzéseket követve, egy prezentációs feladaton keresztül hasonlítja össze egy eszközcentrikus és egy tudástranszfer alapú megoldást, kitérve az egyes megoldások előnyeire és hátrányaira.

**Kulcsszavak:** digitális kultúra, prezentáció, számítógépes gondolkodás, tudástranszfer

## 1. Digitális kultúra

2020 szeptemberében az 5. és 9. osztályokban úgy indult meg az informatikaoktatás – a 2020-as Nemzeti alaptanterv alapján (NAT 2020) [1] *Digitális kultúra* néven –, hogy nem ismertek az előfeltételek, az előzetes ismeretek. Az 5. osztályos tankönyvhöz [2] hiányoztak a 3–4., a 9. osztályoshoz [3] pedig a 3–4. és a 6–8. osztályos tankönyvek. A 2021/2022-es tanév kezdetére megjelent a 6. [4] és 10. [5] osztályos tankönyv, de továbbra is hiányoznak az előzmények. Ilyen indítással rendkívül nehéz a számítógépes gondolkodásról, a számítógépes problémamegoldásról beszélni, mivel ezen képességek fejlesztésének alapfeltétele a tudástranszfer elemek aktiválása. Megfelelő dokumentumok hiányában a tanárra van bízva, hogyan tudja beépíteni a tudástranszfer elemeket a tanulási-tanítási folyamatba az oktatás hatékonyságának növelése érdekében. Fontos megjegyezni, hogy jelen tanulmány írásakor (2021. november) még mindig hiányoznak a 3–4., a 7–8. és a 11. osztályos tankönyvek.

A hivatalos állásfoglalás szerint, tankönyvek hiányában, a tantervi követelmények feldolgozásához a kerettantervek nyújthatnak segítséget [6]–[8], amelyek azonban, hasonlóan a korábbi kerettantervekhez, súlyosan terheltek a túlságosan általános kifejezésekkel [9], így nem adnak elegendő támaszpontot a tananyag önálló, tanári feldolgozásához. További segítség lehet az érettségi követelmények ismerete [10], amelyhez 2021 októberében érkezett meg az első mintafeladatsor [11].

Jelen keretek között a *Digitális kultúra* tantárgy prezentáció témakörének kritikus elemzéséhez és tanórai feldolgozásához adunk javaslatokat. Megvizsgáljuk, hogy a 3–4. osztályos kerettanterv *A digitális eszközök használata* című témakörben milyen tevékenységeket javasol [6], majd ezt követően az 5. osztályos tankönyv [2] *Bemutatókészítés, multimédiás elemek készítése* fejezetének egy tankönyvi feladatánál ötleteket adunk a tantárgyon belüli és az interdiszciplináris tudástranszfer elemek aktiválására.

## 2. Digitális kultúra kerettantervek

### 2.1. Óraszámok

A NAT 2020 [1] alapján a *Digitális kultúra* tantárgy alsós, felsős és középiskolás kerettantervei adják meg a tankönyvi és a tanórai feldolgozás alapjait. A NAT 2020 egyik pozitív hozadéka, hogy minden tantárgy esetén megjelenik a digitális eszközök alkalmazása az adott tantárgy problémamegoldási stratégiájában. Egy további fontos eleme a NAT 2020 dokumentumnak, hogy megemeli az informatikaórák számát, így 3–8. és 10. osztályban heti 1 órát, 9. és 11. osztályban heti 2 órát rendel a tantárgyhoz [1, 299]. Érdemes azonban megjegyezni, hogy ez az óraszám továbbra is lényegesen elmarad a 3R tantárgyak – Reading, wRiting, aRithmetics – óraszámától [1, 299]. Az óraszámok összehasonlítása és a tantárgy indításának éve azért játszik fontos szerepet, mert Wing 2006-os cikkében egyértelműen megfogalmazza [12], hogy a számítógépes gondolkodás meg kell jelenjen a negyedik alapképességként, tehát legalább olyan időkeretben kellene tanítani, mint a 3R tárgyakat.

Az óraszámok vizsgálata során érdemes megemlíteni az érettségi felkészítésre előírt órakereteket is. A 100/1997 kormányrendelet alapján [13, 6. § (4)] a középszintű érettségire felkészítéshez az iskolának legalább 138 órát kell biztosítania, míg az emelt szint esetén legalább 276 órát. A NAT 2020 alapján a középszintű felkészítéshez szükséges gimnáziumi órakeret elegendő, ugyanakkor az emelt szintű messze elmarad a megadott értéktől. Ennek biztosítása további terheket ró az iskolákra a szabadon választható órák terhére.

Nem hagyhatjuk továbbá figyelmen kívül azt sem, hogy a digitális kultúra óraszámok emelését nem előzte meg tanárképzés és/vagy tanártovábbképzés, így a jelenleg is tanárhiánnyal küszködő tantárgy gondjai várhatóan nem fognak enyhülni. A tanárhiány enyhítésére a Debreceni Egyetem indított egy egyéves *Információs- és kommunikációs technológiák (IKT)* című szakirányú továbbképzést, amellyel valamelyest emelhető a nemszakképzett, de informatikához affinitást érző tanárok felkészülési szintje.

### 2.2. Kerettanterv 3–4. osztály

A 3–4. osztályos kerettanterv [6] prezentációhoz leginkább köthető témája az *Alkotás digitális eszközökkel* című fejezet, amely a képfeldolgozásra koncentrálna. A kerettantervi leírás alapján, azonban nem dönthető el, hogy mit tudnak a gyerekek az 5. osztály megkezdésekor. A javasolt tevékenységek, fejlesztési feladatok és ismeretek között található teljesen értelmezhetetlen kifejezések is, pl.: „Az adott célnak megfelelő digitális produktumok létrehozása önállóan, illetve projekt keretében.”, „Képes dokumentum módosítási lehetőségeinek ismerete és alkalmazása”, vagy túlságosan általános feladatok, pl.: „Az iskolai feladatoknak és az egyéni érdeklődésnek megfelelő rajz készítése digitális eszközökkel.”

Ugyanakkor nem hagyhatjuk figyelmen kívül, hogy a 3–4. osztályokban még fontosnak ítélt tevékenységek egyetlen további évfolyamon sem jelennek meg explicit módon:

- a dokumentumok módosítása,
- az osztálytársak dokumentumainak értékelése,
- az alkalmazott megoldások értelmezése,
- mérlegelés, indoklás az adott probléma megoldása során megvalósított digitális eszközhasználattal kapcsolatban.

Annak ellenére hiányoznak ezek a feladattípusok a felsőbb osztályokból, hogy a digitális dokumentumok egyik legnagyobb előnye a papíralapúakkal szemben, a módosíthatóság. Ezen a ponton jelenne meg az egyik legfontosabb tudástranszferencia, a helyesen szerkesztett dokumentum definíciójának alkalmazása, amely minden digitális szöveges dokumentumra egységesen érvényes [14]–[16]. Kiemelten fontos lenne ezen tevékenységek beiktatása és bővítése valamennyi évfolyamon, mivel a 3–4. évfolyamon szerzett ismeretek, elemzési szempontok nem szolgálják ki a felsőbb évfolyamokat.

### 2.3. Kerettanterv 5–8. osztály

Az 5–6. és a 7–8. osztályos kerettantervek alapján 8, illetve 6 óra jut a *Bemutatókészítés* fejezetre. Ezen dokumentumok alapján az 5–6. és a 7–8. osztályos tanulási eredmények között nincs különbség, a két leírás szóról szóra megegyezik. A fejlesztési feladatok és ismeretek esetében is alig van eltérés, hasonlóan a fogalmakhoz (**1. táblázat**). Annak ellenére, hogy a fejlesztési feladatok és ismeretek, valamint a fogalmak nagyban megegyeznek, a javasolt tevékenységek esetében az ismétlődések mellett megjelennek új tevékenységek is (**2. táblázat**).

5–6. osztály	7–8. osztály
Fejlesztési feladatok és ismeretek	
Szöveget, képet tartalmazó prezentáció létrehozása, formázása, paramétereinek beállítása.	Szöveget, <b>táblázatot, ábrát</b> , képet, <b>hangot, animációt, videót</b> tartalmazó prezentáció létrehozása, formázása, paramétereinek beállítása.
	A mondandóhoz illeszkedő megjelenítés
A bemutató objektumaira animációk beállítása.	Automatikusan és az interaktívan vezérelt lejátszás beállítása a bemutatóban.
Fogalmak	
prezentáció, animáció, dokumentumformátum, csoportmunka eszközei, lényegkiemelés, információforrások etikus felhasználása	prezentáció, <b>multimédiás objektum</b> , dokumentumformátumok, csoportmunka eszközei, lényegkiemelés, <b>dokumentum belső szerkezete</b> , információforrások etikus felhasználása

**1. táblázat:** Bemutatókészítés: Eltérések az 5–6. és a 7–8. osztályos kerettantervekben a fejlesztési feladatok és ismeretek és a fogalmak között.

5–6. osztály	7–8. osztály
Minta alapján bemutató létrehozása, paramétereinek beállítása	
Feladatleírás alapján prezentáció szerkesztése	
Prezentáció készítése kiselőadáshoz (a digitális kultúrához, más tantárgyakhoz, az iskolai élethez, hétköznapi problémához kapcsolódó feladat)	Prezentáció készítése kiselőadáshoz (a digitális kultúrához, más tantárgyakhoz, az iskolai élethez, hétköznapi problémához kapcsolódó feladat)
Bemutató készítése projektmunkában végzett tevékenység összegzéséhez, bemutatásához, a megfelelő szerkezet kialakításával, az információforrások etikus használatával	Bemutató készítése projektmunkában végzett tevékenység összegzéséhez, bemutatásához, a megfelelő szerkezet kialakításával az információforrások etikus használatával
	Tájékoztató vagy reklámcélú, automatikusan ismétlődő, animált bemutató készítése
	Rövid rajzfilm készítése prezentációkészítő alkalmazással
	Elkészített prezentáció megjelenítése többféle elrendezésben, mentése különböző formátumokba

**2. táblázat:** Bemutatókészítés: Eltérések az 5–6. és a 7–8. osztályos kerettantervekben a javasolt tevékenységek között.

### 3. Bemutatókészítés – 5. osztályos tankönyv

Az 5–6. osztályos kerettanterv alapján a rendelkezésre álló 12 óra alatt kellene megvalósítani a tankönyv 27 oldalas tömény feladatleírását [2, 35–62], amely a bemutatókészítésen túl magába foglalja a multimédiás elemek készítése – képszerkesztés – témakört is. Összességében, 28–30 feladatot tartalmaz ez a tankönyvi rész. Tekintve egy átlagos 5. osztályos gyereket, akinek még a mentések is gondot jelentenek, szinte lehetetlen feladat elé állítják a tanárokat a tankönyvírók.

Azon túl, hogy ennyi feladatot nem lehet a bemutatókészítésre maradt 8 óra alatt megoldani (ami valójában 7, mert legalább egy számonkérést is be kell iktatni), a tankönyvi leírás több szempontból is hiányos, valamint olyan példákat mutat, amelyek helytelen gyakorlathoz, tanulói megoldásokhoz vezethetnek. Ezen problémák közül említenek néhányat a következő alfejezetek.

#### 3.1. Elrendezések, diaminták vs. szövegdobozok

A PowerPoint prezentációk kiválóan alkalmasak a stílusok bevezetésére és annak gyakorlására, hogy a stílusok módosítása hogyan befolyásolja a diák megjelenését. Az egyik legfontosabb tudástranszferlem, mivel a szövegszerkesztésnél, majd később a weblapszerkesztés során kiválóan lehet(ne) erre az ismeretre támaszkodni a stílusok és a stíluslapok oktatásánál.

Az 5. osztályos tankönyv meg sem említi az elrendezéseket, valamint azt, hogy mit csináljunk a felkínált elrendezések helyőrzőivel. A szöveges tartalmakat kizárólagosan szövegdobozokba kerülnek, ami nem segíti a stílusok megismerését, sokkal inkább helyet adva a barkácsolásnak. A tankönyvi magyarázatnak megfelelően, „*A bemutatókészítőkben a szöveg szövegdobozba kerül.*” [2, 38]. Igen, odakerül, ha odaírjuk, egyébként nem.

PowerPointban, hasonlóan más szoftverekhez, amennyiben az egységes megjelenés preferáljuk a szövegdobozok használata és módosítása rendkívül körülményes, szemben az elrendezésmintákon beállított helyőrzőkkel.

#### 3.2. Nyelvtani szabályok

Teljesen hiányzik a tankönyvi feladatokból a nyelvtani szabályok betartásához köthető tudástranszferlemek említése és alkalmazása. Ezek az ismeretek a magyar és az idegen nyelvi órákról érkezhettek, ahol a gyerekek már kisiskolás kortól tanulják a nyelvtani szabályokat. Annak ellenére azonban, hogy már az 1–2. osztályos magyar kerettanterv megemlíti a tanulási eredmények között, hogy a tanuló digitális forrásokat használ, a tankönyvek figyelmen kívül hagyják a kerettantervet, és kizárólagosan a kézzel írott dokumentumokra koncentrálnak.

Történik mindez annak ellenére, hogy kiváló alkalmazási lehetőség lehetne a prezentációs szövegek helyességének ellenőrzése, a nyelvi beállítások módosítása, előkészítve ezzel a hosszabb szövegek helyességének ellenőrzését. A nyelvtani szabályok valódi digitális szövegeken történő alkalmazásával egy egész hosszú tudástranszferláncot tudnánk felépíteni, amivel elkerülhető lenne az értelmetlen feladatok és szövegek alkalmazása a *Digitális kultúra* tantárgyban [3] [4].

#### 3.3. Képek kezelése

A képek formázása és átalakítása a kerettantervek [6] [7] és az 5. osztályos tankönyv alapján ismertnek tekinthető [2], így teljesen meglepő, hogy a prezentációban felhasznált képek esetén nem kerülnek aktiválásra a képekhez köthető tudástranszferlemek. A tankönyv egyik komoly hiányossága a mintaként adott képek vágásának megkerülése, a képszerkesztő programok kihagyása a folyamatból. A prezentációs feladatoknál a tankönyv kétféleképpen kerüli meg ezt a problémát. Egyik esetben meg sem említi, hogy szükség van a kép megvágására [2, 48], annak ellenére, hogy a mintán látszik, máskor pedig nagyon „cselesen” lelógatja a kép szélét a diáról [2, 49].

A képek vágása és szerkesztése a prezentációs feladatoktól függetlenül, a *Képek vágása*, a *Képek szétvágása* és a „*Képek vágása színezettség alapján*” alfejezetekben jelenik meg, figyelmen kívül hagyva, hogy ezekre a műveletekre már a korábbi feladatokban is szükség lett volna.

A képek tárgyalásánál, azonban, mindenképpen meg kell említeni, hogy az 5. [2] és a 9. osztályos könyvekben [3] az ábrák minősége vállalhatatlanul gyenge (1. ábra, bal; 2. ábra, fecskek). Különösen visszatetsző a tankönyvi ábrák gyenge minősége abban az esetben, amikor a tárgyalt témakör éppen a képek kezelése [2].

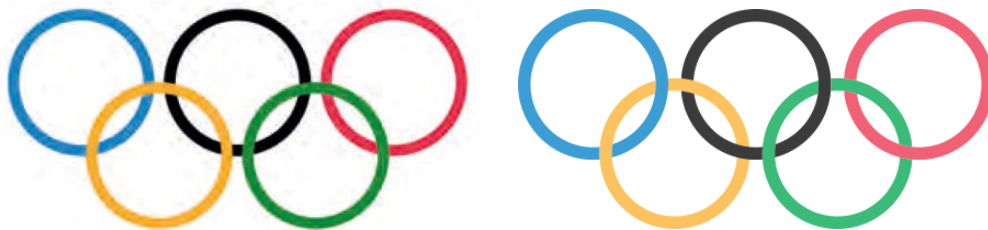
### 3.4. Szellemi tulajdon, forrásmegjelölés

Valamennyi kerettantervben hangsúlyosan jelennek meg az „*Az információforrások etikus felhasználásának kérdése*”. Mindezt figyelmen kívül hagyva, az 5. osztályos tankönyv használ olyan képeket, amelyeket szerzői jogok védenek. Ilyen képek például az olimpia ötkarika [2, 43] (**1. ábra**) és a Vasarely kép [2, 44]. Teljesen hiányzik az 5. osztályos *Digitális kultúra* tankönyvből a szellemi tulajdon-védelem céljának tárgyalása [17], ami a különböző területeken történő szellemi tevékenységből született szellemi tulajdonok erkölcsi és jogi védelme, illetve azok anyagi elismerésének biztosítása.

A forrásmegjelölés a következő sarkalatos pontja a képek felhasználásának. A tankönyv alapján „*Ha nem a saját képeinket használjuk, a műveinkben meg kell jelölnünk, hogy honnan származnak, azaz mi a képek forrása. Lehetséges megoldás, hogy az utolsó „rendes” dia után nyitunk egy új diát. Ide kiírhatjuk, hogy ki készítette a műünk alapjául szolgáló képeket. Ha az internetről töltöttünk le valamit, írjuk ki, hogy melyik web-címről és mikor töltöttük le*” [2, 57].

Annak ellenére, hogy a tankönyv felhívja a tanulók figyelmét a források megnevezésére, sem a tankönyv, sem a tankönyvi prezentációk nem tartalmaznak egyetlen forrásmegjelölést sem.

A képek felhasználásánál azt is érdemes lenne figyelembe venni, hogy az eredeti képeket ne rontsuk el, ahogy az olimpia ötkarika esetén ez megtörtént (**1. ábra**). Ugyanakkor a Vasarely kép esetén még a kép címét sem adja meg a tankönyv, így rendkívül nehéz ellenőrizni, hogy valóban Vasarely képről van-e szó vagy sem (internetes képkeresés nem adott eredményt).



**1. ábra:** A tankönyvi (bal) [2, 43] és a valódi (jobb) [19] olimpiai ötkarika összehasonlítása.

### 3.5. Tipográfia

A tipográfiai ismeret is nagyon szorosan köthető a tantárgyak közötti tudástranszferhez, mivel sokkal inkább tartozik a *Vizuális kultúra* tantárgyhoz [18], mint az informatikához. A tipográfia egyik legnagyobb tématerülete a betűtípusok. A betűtípusok kiválasztásánál ugyanazt a hibát követi el az 5. osztályos tankönyv [2], mint a 6. [4] és a 9. osztályos [3] tankönyvek a szövegszerkesztés témakörnél: „*Válasszunk jól olvasható és érdekes, de nem túldíszített betűket!*”. Figyelmen kívül hagyja, hogy a magyar nyelvű szöveges dokumentumok – különösen igaz ez a prezentációkra – egyik legnagyobb hibája, hogy olyan betűtípust választ a szerző, majd ennek mintájára a tanuló, amelyből hiányoznak az ékezetes karakterek. Az 5. osztályos tankönyv mintafeladata alapján [2, 38] készített **2. ábra** (bal) arra mutat példát, hogy mi történik helytelen betűtípusválasztás esetén. A mintán, a tankönyvi dia szöve-

gét kiegészítettük ékezetes karaktereket tartalmazó kifejezésekkel, ilyenkor a fontkészletből hiányzó betűk a diamintán beállított betűtípussal kerülnek kiegészítésre.

A következő gyakori tipográfiai hiba az 5. osztályos tankönyvben, hogy az objektumok elhelyezését próbálkozással és nézegetéssel kellene megoldani. A tipográfiai ismeretek átvétele a *Vizuális kultúra* tantárgyból kiemelten fontos lenne ahhoz, hogy tipográfiai szempontból is helyes prezentációk, a későbbiekben pedig szöveges dokumentumok, készüljenek. A jelenleg érvényes kerettantervek alapján, azonban a *Vizuális kultúra* tantárgyban csak 7–8. osztályra van tervezve tipográfia [17]. Iskolán belül azonban érdemes megfontolni ennek a témakörnek az átmozgatását az 5. osztály elejére, mielőtt megkezdődik a prezentációs témakör. Ezen tantárgyközi kapcsolat részletes elemzése és feldolgozása, azonban túlmutat jelen tanulmány keretein.



**2. ábra:** Hibásan választott betűtípusok, amelyek nem tartalmazzák az ékezetes betűket. A fecske feladat [2, 39] kiegészítve a vízerőmű szóval (bal), az internetről letöltött prezentáció (jobb) [20] és egy weblap (lent) [21] minta alapján.

A tankönyvi tipográfia figyelmen kívül hagyja továbbá, hogy a számozásnál a helyiértékes elrendezést kellene használni. Így a 9. sorszámról a 10.-re váltásnál a tízes az korábbi egyesek alá kerül (3. ábra).

**9.** Állítsunk be lassú, automatikus megjelenést a csillagokra is! Egyszerre, vagy egymást követően jelenjenek meg? Melyik az érdekesebb animáció, melyik áll közelebb a valósághoz?

Tegyük bonyolultabbá az animációt!

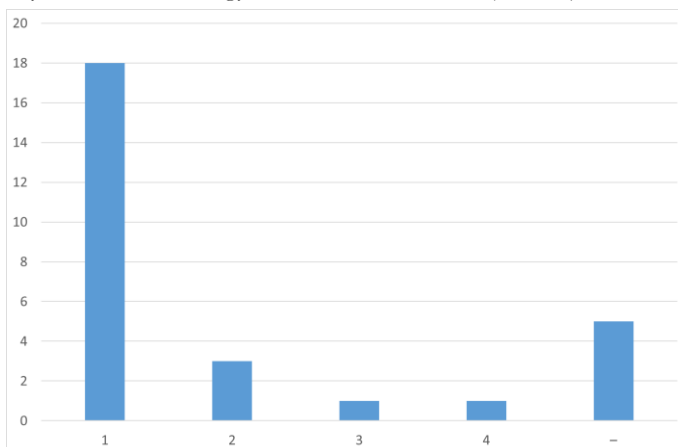
**10.** A felhős háttérre rajzoljunk a bemutató szabad kézi vonal eszközével világoszöld „tájat”! Töltsük ki világoszölddel!

**3. ábra:** Balra igazított számozás az 5. osztályos *Digitális kultúra* tankönyvben [2, 50].

### 3.6. Diák száma

A tankönyvi feladatok többsége nem valódi prezentációt készít, hanem csak egy-egy, szöveggörnyezetből kiragadott diát formázgat. Történik ez annak ellenére, hogy a fejezet elején a következő ígéretet kapjuk: „A legjobb bemutatókészítő alkalmazás mind a mai napig egymás után vetített diákból álló bemutatót készít. Mi is ilyeneket készítünk ebben az évben.” A tankönyv nem említi továbbá, hogy a következő tanévek egyikében sem téma a prezentációkészítés.

A feladatok egyértelműen mutatják azt is, hogy az eszközhasználat megtanítása a cél, és nem a prezentációtervezés, -kezelés és -szerkesztés. A 28 tankönyvi feladat diáinak a számát tekintve az egydiás prezentáció a leggyakoribb, ezt követi az az eset, amikor nem ismerjük a diák számát, és alig találni olyan tankönyvi feladatot, ami egynél több diát tartalmaz (**4. ábra**).



4. ábra: A diaszámok eloszlása az 5. osztályos tankönyv prezentációs feladatai alapján.

#### 4. A lakótorony feladat tudástranszferlemei

Az 5. osztályos tankönyv lakótorony feladatának [2, 48] (**5. ábra**) két lehetséges megoldását mutatjuk be. Az egyik az eredeti tankönyvi kiírás alapján (0 fejezet), míg a másik a tudástranszferlemeiket felhasználva (0 fejezet).

Készítsük el az alábbi diát, figyelve arra, hogy a szöveg minden sora külön szövegdobozba kerüljön! A dia elkészítéséhez használjuk a könyv webhelyéről letölthető lakotorony.tif képet. Amikor elkészültünk, állítsuk be úgy a szövegdobozok animációját, hogy a szövegek alulról kezdve jobbról, egyesével érkezzenek meg!



5. ábra: A lakótorony prezentációs feladat leírása az 5. osztályos tankönyvben.

##### 4.1. Tankönyvi megoldás

A tankönyv teljesen kihagyja a diatervezés fázisát, így nem kerül tárgyalásra, hogy milyen elrendezést/elrendezéseket célszerű használni a mintán megadott tartalomhoz. Felhívja a figyelmet, hogy minden egyes szöveget külön szövegdobozba helyezzünk el (**5. ábra**): „Készítsük el az alábbi diát, figyelve arra, hogy a szöveg minden sora külön szövegdobozba kerüljön?”. Annak ellenéris teszi ezt, hogy ehhez a tartalomhoz sokkal inkább megfelel egy 2 tartalomrész elrendezés, amely három olyan helyőrzőt

tartalmaz – cím, tartalom1, tartalom2 –, ami megfelel az adott tartalomnak. A bal oldali helyőrzőbe beszúrható a kép, míg a jobb oldaliba egy felsorolással a torony részei.

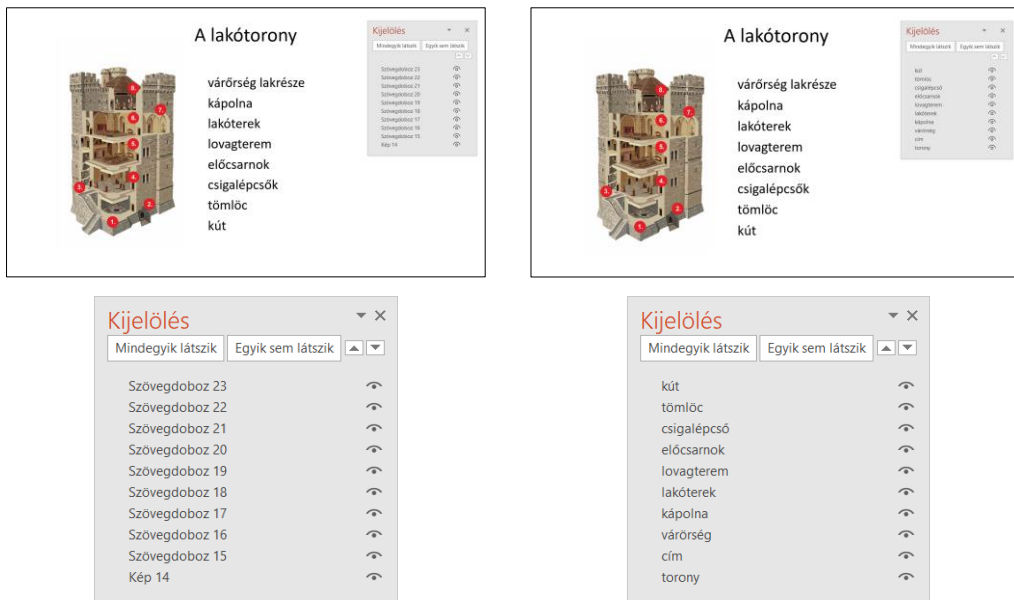
A tankönyvi megoldás egy további hiányossága, hogy nem tesz említést a diatervezésről, így a következő megfontolások kimaradnak a feldolgozásból:

- szövegdobozok mérete, pozícionálása, vízszintes és függőleges igazítása;
- cím szövegdoboz elhelyezése, cím formázása;
- kép elhelyezése, méretezése, körülírása.

A tankönyv figyelmen kívül hagyja továbbá, hogy minden új dokumentum egy *Cím dia* elrendezésű diával indít. Így az sem kerül tárgyalásra, hogy mi történjen ezzel az elrendezéssel, illetve az elrendezéshez tartozó helyőrzőkkel.

A sok-sok szövegdoboz további hátránya, hogy az automatikus elnevezésekkel rendkívül körülmenyes az animációs sorrend beállítása (részletes tárgyalásra a **8. ábra** alapján a 0 fejezetben kerül sor). Ennek megfelelően, a hatékony problémamegoldás mindenképpen szükségessé teszi az objektumok átnevezését (**6. ábra**).

Fontos azonban megjegyezni, hogy a szövegdobozos megoldásnak is lehetne tudástranszfer vonzata, mivel az objektumok ily módon történő kezelése egy kiemelten fontos tudástranszfer elem, ami nagyon sok témakörben, szoftverben megjelenik. Az objektumok alapszintű kezelésének tanítására kiválóan alkalmas lehet a PowerPoint, mivel ebben a környezetben nagyon egyszerű az objektumok megjelenítése, átnevezése. A tankönyvi megoldásban ez a tudástranszfer elem nem kerül említésre.



**6. ábra:** A lakótorony szövegdobozok eredeti elnevezése (bal) és azok átírása beszédes nevekre (jobb).

További nagy hiányossága a tankönyvi leírásnak, hogy nem tartalmaz diszkussziót, nem ad javaslatokat arra, hogyan lehet a leírt megoldást hatékonyan kivitelezni, akár még a szövegdobozokkal is. Az informatikaoktatás hatékonyságának növelése és a számítógépes gondolkodás fejlesztése érdekében kiemelt szerepet kellene kapjon a különböző lehetséges megoldások összevetése az alkalmazói ismeretek tanítása során is. Hasonlóan a programozáshoz, az alkalmazói feladatok megoldása is megkö-



veteli a kapott eredmények és megoldások ellenőrzését, a lehetséges megoldások kipróbálását és összehasonlítását, valamint az általánosítás lehetőségének vizsgálatát, ahogy az megjelenik a 3–4. osztályos kerettantervben [6].

## 4.2. Megoldási javaslat

### 4.2.1. Megjelenés

A dia tervezésénél érdemes egy olyan megoldást keresni, amely a későbbiekben általánosítható. Egy nagyon egyszerű általánosítási lehetőség a dia elkészítése több nyelven vagy hasonló építmények bemutatása.

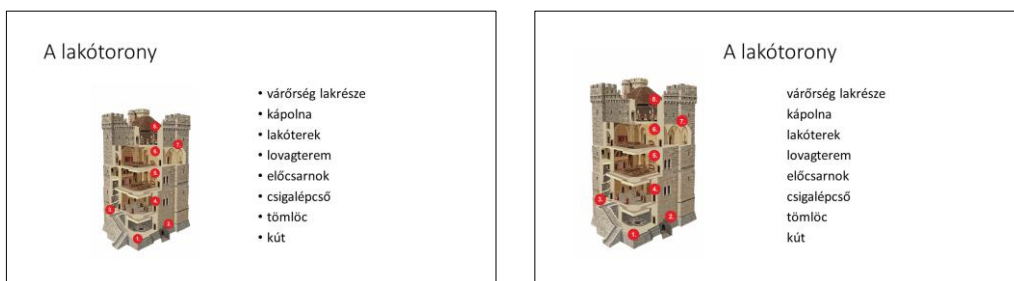
A tankönyvi minta alapján a dia három objektumot tartalmaz: cím a lap tetején, egy kép és egy felsorolás egymás mellett. Ennek az elrendezésnek a megvalósításához a *2 tartalomrész* elrendezés a leginkább célszerű.

Az elrendezésekkel történő megvalósítás előnye, hogy több hasonló tartalmú dia esetén csak egy helyen, a diamintán kell elvégezni a formázási módosításokat, valamint a szöveges tartalmak áttekintése lényegesen egyszerűbb a *Vázlatnézetben*, a helyőrzőket használva. A *2 tartalomrész* elrendezés további előnye, hogy ezzel a megoldással csak három objektum kerül a diára, így nincs szükség a rengeteg szövegdoboz egyéni kezelésére és formázgatására. A három helyőrző átnevezése lényegesen kevesebb időt vesz el, mint a szövegdobozoké.

A *Diaminta* nézetben arra is lehetőségünk van, hogy apró módosításokat hajtsunk végre az elrendezésmintán (0 fejezet). A **7. ábra** az alapértelmezés szerinti és a módosított *2 tartalomrész* elrendezésminta alapján készített laktótorony diákat mutatja.

A tankönyvi minta alapján egyetlen dia létrehozása valóban nem indokolja a stílusok – PowerPointban elrendezések – alkalmazását és módosítását. Érdemes azonban messzebbre gondolkozni, mivel ilyen egyszerű stílusokkal elő lehet készíteni a szöveg-, valamint a weblapszerkesztői stílusok tanítását és alkalmazását, melyek a prezentációtervezésnél jóval bonyolultabbak. Mindez azért kiemelt fontosságú, mert a stílus olyan tudástranszferrelem, amely a digitális dokumentumok hatékony szerkesztésének egyik alapvetése.

A szöveges dokumentumok létrehozásakor érdemes arra is kitérni, hogy az automatikus javítási beállításoknál kapcsoljuk ki, hogy „*A mondat első betűje legyen nagybetű*” és „*A táblázatcellák szövege nagy kezdőbetűvel*”. A laktótorony dokumentum kiválóan alkalmas arra, hogy megmutassuk „*A mondatok első betűje*” kapcsoló használata milyen előnyökkel és hátrányokkal jár. Ez is egy olyan tudástranszferrelem, amely a szövegkezelés tanításához átemelhető.

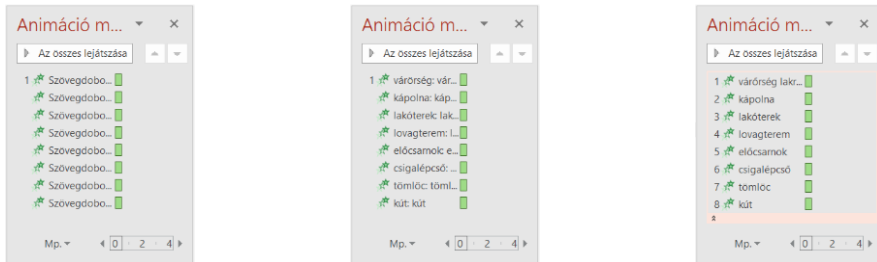


**7. ábra:** A lakótorony tartalom az eredeti (bal) és a módosított *2 tartalomrész* elrendezéssel (jobb).

A tankönyvi mintából kiindulva, a szövegdoboz- (**5. ábra**) és az elrendezésalapú (**7. ábra**, jobb) megvalósítást összehasonlítva, a megjelenésben nincs eltérés. A következő kérdés, hogy az animáció hogyan valósítható meg a kétféle kivitelezésben.

#### 4.2.2. Animáció

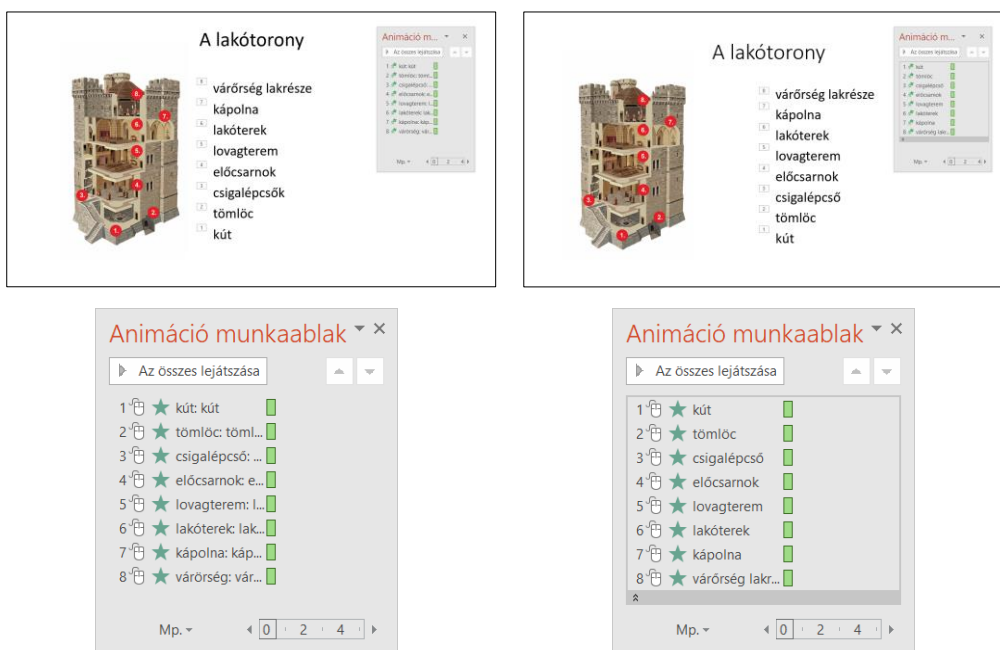
A feladat kiírásának megfelelően, „...állítsuk be úgy a szövegdobozok animációját, hogy a szövegek alulról kezdve jobbról, egyesével érkezzenek meg” [2, 48]. Ez az animáció megvalósítható mind a szövegdobozokkal, mind a szöveges helyőrzővel. Tehát, továbbra sem indokolt a szövegdobozok alkalmazása (8. ábra).



**8. ábra:** A lakótorony szövegek animációja alapértelmezett szövegdobozokkal (bal), átnevezett szövegdobozokkal (közép) és szöveges helyőrzővel (jobb).

A **8. ábra** mutatja a lakótorony szövegek animációját három különböző megoldással:

- A bal oldali minta az alapértelmezés szerinti szövegdozelnevezésekkel, ahol csak annyi látszik az animációs munkaablakban, hogy valamiféle szövegdozokra tettünk animációt.
- A középső minta az átnevezett szövegdozok animációját mutatja. Ezen az ábrán már egyértelműen lehet látni, hogy melyik objektum hol helyezkedik el az animációs sorrendben.
- A jobb oldali minta a helyőrzőben elhelyezett szövegek animációját mutatja, ahol egyértelmű, hogy a szövegek hogyan követik egymást.



9. ábra: A lakótorony animációs sorrendje a módosítások elvégzése után a szövegdobozos (bal) és az elrendezéses (jobb) megoldást alkalmazva.

A két megoldás – szövegdobozok és helyőrző – abban is különbözik egymástól, hogy a szövegdobozok együttes kijelölésével egyetlen animáció kerül az összes objektumra, így egy további lépés szükséges ahhoz, hogy az objektumok egymás után jelenjenek meg (*Kattintásra indul*). A helyőrző esetén minden bekezdés önálló animációt kap, így az animációk indításához további beállításokra nincs szükség.

Az animációs sorrend módosítását mindkét esetben el tudjuk végezni húzással, úgy, hogy a „kút” kerüljön az animációs lista legelejére, míg a „várórség lakrésze” a végére. Mindkét megoldást a 9. ábra mutatja.

#### 4.2.3. Elrendezésminta módosítása

A lakótorony feladat tartalmának leginkább a 2 *tartalomrész* elrendezés felel meg. Ezt az elrendezést alkalmazva a 7. ábra bal oldali mintáján bemutatott eredményhez jutunk. Ahhoz, hogy az eredeti kiíráshoz jobban illeszkedő megjelenítést kapjunk, érdemes a 2 *tartalomrész* elrendezésminta alapján létrehozni egy újabb elrendezésmintát, majd azt a feladat alapján formázni (7. ábra, jobb).

Ezzel a megoldással elsősorban a stílusok létrehozását és alkalmazását tanuljuk és gyakoroljuk, ugyanakkor további tudástranszferelemek is megjelennek: másolás, átnevezés, felsorolás és tipográfiai megfontolások. Ezek a tudáselemek az informatikaoktatás során számos témakörben megjelennek, csak a környezet változik. Kérdés, hogy a tanulók fel tudják-e ismerni, hogy ezekkel a problémákkal már találkoztak korábban, és igazából csak tudástranszferaktíválás történik. A korábbi kerettantervek alapján történő tanítás tapasztalatai azt mutatják, hogy a másolás és átnevezés műveletek izoláltan jelennek meg *Az informatikai eszközök használata*, míg a felsorolás *Az alkalmazói ismeretek* fejezetben, a tipográfiai megfontolások pedig szinte sehol.

A NAT 2020 alapján készült kerettantervekben a prezentációkészítés megelőzi a szövegkezelést, így hivatalosan elsőként a prezentációszerkesztés során találkozhatnak a tanulók a tipográfiával és a

felsorolásokkal. A 3–4. osztályos kerettanterv alapján, tankönyv hiányában nem tudjuk megjósolni, hogy mely témakörökben kerül bevezetésre és gyakorlásra a másolás, átnevezés. Azt azonban mindenképpen szem előtt kell tartanunk, hogy gyakorlásra van ahhoz szükség, hogy a hosszútávú memóriában tárolásra kerüljenek az egyes tudáselemek. Ennek megvalósításához pedig az egyik lehetséges megoldás a tudástranszferaktiválás.



**10. ábra:** Az eredeti 2 *tartalomrész* elrendezés minta (bal) és az ez alapján készült *torony* elrendezés minta (jobb).

A *torony* elrendezés minta a 2 *tartalomrész* elrendezés minta alapján készült az alábbi formázásokkal (**10. ábra**):

- cím középre igazítása
- 2–5 felsorolásszintek törlése
- 1-es szinten felsorolás kikapcsolása

Valamennyi végrehajtott formázás átemelhető tudáselemeket tartalmaz, így nem csak az elrendezés minta formázása során alkalmazható, hanem más típusú szöveges dokumentumok formázásánál is.

## 5. A kiscső feladat tudástranszferlemei

### 5.1. Feladat megfogalmazása

A tudástranszferelemekhez és ezek aktiválásához szorosan köthető a feladat megfogalmazásának módja. Az 5. osztályos *Digitális kultúra* tankönyvben kiemelten fontos szerepet kap az eszközök használatának tanítása leírás alapján. Nem kap ugyanakkor elegendő figyelmet a valódi problémamegoldás. Erre mutat példát az alábbi rövid elemzés, ahol a kiscső feladat leírását vizsgáljuk meg (**11. ábra**, **12. ábra**).



11. ábra: A kiscső feladat leírása a 10. lépéstől [2, 50].

## 5.2. Tankönyvi megoldás

A kiscső feladat leírása alapján a 9. és a 10. lépés között azzal ijesztgetjük a tanulókat, hogy valami nagyon bonyolult animáció következik: „*Tegyük bonyolultabbá az animációt!*”. Ugyanakkor semmi nem indokolja ezt a riadalmat. Való igaz, hogy a leírás alapján nem tudjuk meg, hogy pontosan mi a feladat célja, csak követjük a szakácskönyvet (computer cooking), és majd történik valami. A tankönyv szellemében, a kiscső feladat 10–12. lépéseihez hasonlóan, az 1–9. lépéseik is nagyban fókuszálnak az eszközhasználatra (12. ábra). Ezzel a megoldással a tanulókat sokkal inkább parancsüzemmódba állítva, mintsem a feladat megtervezésére helyezve a hangsúlyt.

### Kiscső

Látványos, bár nagyon egyszerű bemutatót készítünk az ismert csillagképről. Valójában bármelyik csillagkép ábrázolható így. Rendelkezésre áll a könyv webhelyéről letölthető felhok.jpg fájl, de letölthetünk magunknak egy nekünk jobban tetszőt is az internetről.

1. Állítsunk be a diának felhős hátteret!
2. Keressünk képet az interneten egy nekünk kedves csillagképről! Szúrjuk be a diára, és nagyítsuk ki!

5. Rajzoljunk egy, az egész diát eltakaró fekete téglalapot!
6. Küldjük a téglalapot a csillagok mögé!

A feladat leírása 4, egymástól teljesen független állítást tartalmaz.

Nem tudjuk, hogy miért van szükség a felhős háttérre és azt sem, hogy az internetről letöltött csillagképre.

Nem tudjuk, hogy miért van szükség a fekete téglalpra.

12. ábra: Eszközcentrikus feladatleírás a kiscső feladathoz.

Óriási hiányossága továbbá a tankönyvnek, hogy nem készült, legalább a tanárok számára letölthető melléklet, amely tartalmazza a megoldásokat. Különösen fontos lenne a megoldások közlése a kiscsőhöz hasonló animált feladatoknál, amikor a képi megjelenítés nem elegendő az ellenőrzéshez. A tankönyv nem tartalmazza továbbá az animációs algoritmusokat szemléltető *Animációs munkaablakokat*, ami nagyban segítené az egyszerű algoritmusok megépítését és elemzését.

## 5.3. Megoldási javaslat

Egy egészen más típusú gondolkodásmódot igényel a feladat megoldása, ha nem a kattintgatások sorozatának követését várjuk el a tanulóktól, hanem átfogalmazzuk a feladatot: „*Készítsünk olyan animációt, amelyben a nappalok és az éjszakák váltakozva jelennek meg!*”

Az így átfogalmazott feladat megköveteli a prezentáció tervezését. Többek között döntünk kell arról, hogy

- milyen objektumokra van szükség a nappali és az éjszakai képhez;
- hol helyezkedjenek el az objektumok;
- milyen legyen ezen objektumok formázása;
- mi legyen a rétegek sorrendje.

Az animáció tervezése a következő megfontolásokat foglalja magába:

- mi legyen az első (kiindulási) és a második fázis;
- milyen objektumok jelenjenek meg az első fázisban;
- milyen objektumok tűnjenek el az első fázisban;
- milyen objektumok jelenjenek meg a második fázisban;
- milyen objektumok tűnjenek el a második fázisban;
- melyik animáció induljon kattintásra és melyik az előtte levővel.

Arról is döntenünk kell, hogy a vetítés során előre/hátra mozgunk az animációban (pl. a jobbra/balra kurzormozgató nyilakkal) vagy beállítjuk a vetítésnél, hogy az Esc billentyű lenyomásáig folyamatosan ismételhető legyen az animáció.

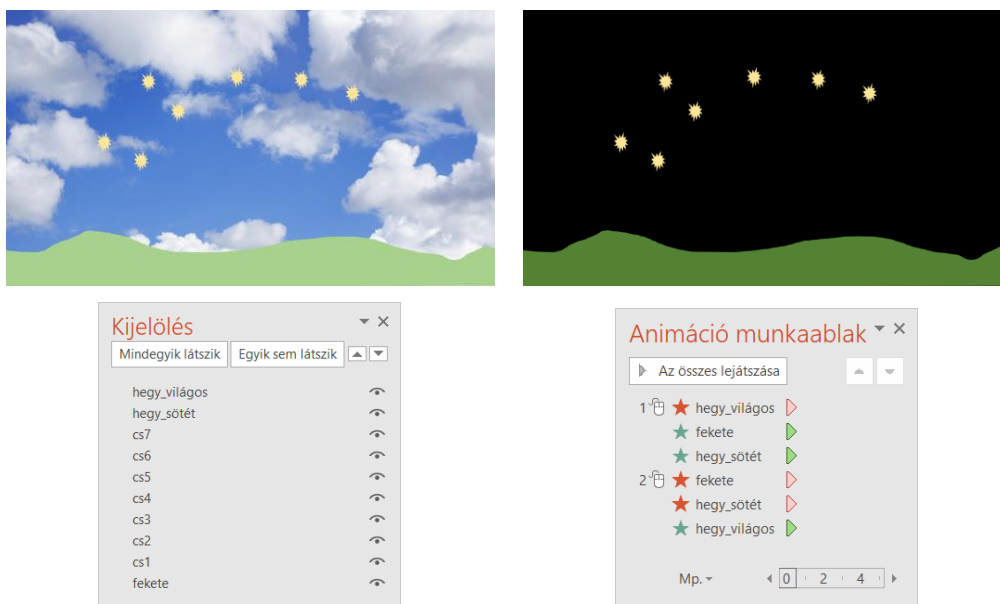
Ezen utóbbi tervek alapján egy lehetséges megoldást mutat a **13. ábra**. A felhős háttérrel a diamintán helyeztük el, így ez fixen a háttérben marad. A dián a leghátsó rétegre helyeztük el a fekete téglalapot, majd erre a csillagokat, és végül a két hegyet (*hegy\_világos* és *hegy\_sötét*). A hegyek esetén döntenünk kellett arról, hogy melyik helyezkedjen el a magasabb (elülső) rétegen. A réteg elhelyezése attól függ, hogy a nappali vagy az éjszakai képpel fogunk indítani. A mellékelt megoldásban a nappali képpel indítunk.

Ahhoz, hogy a nappali képet leváltsa az éjszakai, el kell tűnjön a világos hegy és meg kell jelenjen a fekete téglalap a sötét hegygel. Ahhoz, hogy a nappali kép leváltsa az éjszakai el kell tűnjön a fekete téglalap és a sötét hegy, és meg kell jelenjen a világos hegy.

Az animáció két fázisát mutatja a **13. ábra** felső két mintája, a dia objektumait és az animációs sorrendet pedig az alsó két minta.

Egy jól megtervezett prezentáció és animáció megvalósítása, helyességének ellenőrzés és a megoldás diszkussziója lényegesen hatékonyabban valósítható meg, mint egy olyan szakácskönyv (computer cooking) követése, amelyben nem tudjuk pontosan, hogy mit szeretnénk elérni.

A kiscgöncöl, a tankönyvi feladatok többségéhez hasonlóan, egyértelműen mutatja, hogy egy remek ötletet hogyan tudunk elrontani. Hiába a jó feladatötlet, ha a tankönyvi megoldás nem ad teret a valódi problémamegoldáshoz. Csak és kizárólag az eszközökön van a hangsúly, teljesen figyelmen kívül hagyva az eszközök mögött meghúzódó tartalmakat.



13. ábra: A kisgöncöl feladat egy lehetséges megoldása az előre megfogalmazott prezentáció és megvalósítási tervek alapján. A megoldáshoz használt csillagkép a forrásban megadott URL-ről érhető el [22].

## 6. Összegzés

Jelen tanulmányban azt vizsgáltuk meg, hogy a NAT 2020 alapján készített digitális kultúra kerettantervek és tankönyvek hogyan támogatják a tudástranszferelemek aktiválását, valamint hogyan próbálnak elszakadni a korábbi felületcentrikus gyakorlatoktól.

Az elemzés egyértelműen mutatja, hogy a tudástranszferelemek aktiválásának egyik nagy akadályozója éppen a tankönyv, a tankönyvek megjelenésének sorrendje, a tankönyvi feladatleírások. A 2021/2022-es tanévben, annak ellenére, hogy már második éve úgy folyik a NAT 2020 alapján történő informatikaoktatás, nem ismerjük az előzményeket, nem tudjuk, hogy a tanulók milyen tudáselemeket hoznak magukkal a korábbi évfolyamokról. Várhatóan ezzel magyarázható az is, hogy a 9. osztályos tankönyv megismétli a 6. osztályos szövegszerkesztési tananyagot, és bátran átveszi annak hibáit is. Ez lehet annak is a magyarázata, hogy egyetlen tankönyvi fejezetben sem található utalás további informatikai témakörökre. Az egyes fejezetek teljesen izoláltan kerülnek tárgyalásra.

A tudástranszferelemek hiánya azonban nem csak az egyes fejezetek között jelenik meg, hanem az egyes feladatok tartalmában és megoldási módjában is. A jelen tanulmányhoz kiválasztott feladatok is figyelmen kívül hagyják a lehetséges tudástranszferelemeket. Nem kerül megfogalmazásra, hogy milyen korábbi ismereteket lehet felhasználni, valamint az sem, hogy milyen hatása lehet a prezentációkészítés során szerzett tudáselemeknek a későbbi tanulmányok során.

A kiválasztott feladatokhoz adott tankönyvi megoldások legnagyobb problémája, hogy kimarad mind a prezentáció, mind a megvalósítás tervezésének folyamata, nem kerülnek említésre a stílusok (elrendezések), azok hatékony alkalmazása és módosíthatósága. Ezen kompetenciák fejlesztése azért lenne kiemelten fontos, mert egyrészt segíthetnek a helyesen szerkesztett prezentációk létrehozásában, ugyanakkor a stílusokról szerzett ismeretek átemelhetők lennének más típusú szöveges dokumentumok – pl. szövegszerkesztés és weblapszerkesztés – tervezéséhez és formázásához.

Az elemzések alapján összegzésként megfogalmazhatjuk, hogy érdemes lenne megfontolni a tankönyvi feladatok oly módon történő átdolgozását, hogy azok támogassák a tudástranszferelemek aktiválását, a tanulói tervezéseket, gondolkodást, segítve ezzel a hatékony számítógépes problémamegoldást, a tanulók számítógépes gondolkodásának fejlesztését.

## Irodalom

- 5/2020. (I. 31.) A Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról szóló 110/2012. (VI. 4.) Korm. rendelet módosításáról.  
[http://ofi.hu/sites/default/files/attachments/mk\\_nat\\_20121.pdf](http://ofi.hu/sites/default/files/attachments/mk_nat_20121.pdf) (utoljára megtekintve: 2021. 11. 01.)
- A. Lénárd, A. Abonyi-Tóth, N. Turzó-Sovák, P. Varga: Digitális kultúra 5. Oktatási Hivatal. (2020)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG05TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG05TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
- P. Varga, K. Jenciné Horváth, Z. Reményi, Cs. Farkas, I. Takács: Digitális kultúra 9. Oktatási Hivatal. (2020)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG09TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG09TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
- A. Abonyi-Tóth, Cs. Farkas, N. Turzó-Sovák, P. Varga: Digitális kultúra 6. Oktatási Hivatal. (2021)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG06TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG06TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
- A. Abonyi-Tóth, Cs. Farkas, K. Jenciné Horváth, Z. Reményi, T. Tóth, P. Varga: Digitális kultúra 10. Oktatási Hivatal. (2021)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG10TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG10TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
- Oktatási Hivatal: Kerettanterv az általános iskola 3–4. évfolyamára. Digitális kultúra. (2020)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/kerettanterv/Digitalis\\_kultura\\_A.docx](https://www.oktatas.hu/pub_bin/dload/kozoktatas/kerettanterv/Digitalis_kultura_A.docx). (2020) (utoljára megtekintve: 2021. 11. 01.)
- Oktatási Hivatal: Kerettanterv az általános iskola 5–8. évfolyamára. Digitális kultúra. (2020)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/kerettanterv/Digitalis\\_kultura\\_F.docx](https://www.oktatas.hu/pub_bin/dload/kozoktatas/kerettanterv/Digitalis_kultura_F.docx). (2020) (utoljára megtekintve: 2021. 11. 01.)
- Oktatási Hivatal: Kerettanterv a gimnáziumok 9–12. évfolyamára. Digitális kultúra. (2020)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/kerettanterv/Digitalis\\_kultura\\_K.docx](https://www.oktatas.hu/pub_bin/dload/kozoktatas/kerettanterv/Digitalis_kultura_K.docx). (utoljára megtekintve: 2021. 11. 01.)
- T.K. Nagy: Az informatika kerettanterv elemzése. 3th International Interdisciplinary Conference 2018.  
[https://detep.unideb.hu/sites/default/files/upload\\_documents/kotet\\_interdisz\\_3.pdf](https://detep.unideb.hu/sites/default/files/upload_documents/kotet_interdisz_3.pdf) (utoljára megtekintve: 2021. 11. 01.)
- Digitális kultúra. Részletes érettségi vizsgakövetelmények. Feltöltés dátuma: 2021. 07. 16. (2021)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/dig\\_kult\\_2024\\_e.pdf](https://www.oktatas.hu/pub_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/dig_kult_2024_e.pdf) (utoljára megtekintve: 2021. 11. 01.)
- Oktatási Hivatal: Minta feladatok a 2024. május-júniusi vizsgaidőszaktól érvényes vizsgakövetelmények (2020-as Nat-ra épülő vizsgakövetelmények) szerint. Feltöltés dátuma: 2021. 10. 12. (2021)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok\\_2024/digkult\\_kozep\\_gyakorlati\\_minta.pdf](https://www.oktatas.hu/pub_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok_2024/digkult_kozep_gyakorlati_minta.pdf)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok\\_2024/digkult\\_kozep\\_gyakorlati\\_minta\\_-\\_melleklet.zip](https://www.oktatas.hu/pub_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok_2024/digkult_kozep_gyakorlati_minta_-_melleklet.zip)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok\\_2024/digkult\\_emelt\\_gyakorlati\\_minta.pdf](https://www.oktatas.hu/pub_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok_2024/digkult_emelt_gyakorlati_minta.pdf)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok\\_2024/digkult\\_emelt\\_gyakorlati\\_minta\\_-\\_melleklet.zip](https://www.oktatas.hu/pub_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2024/mintafeladatok_2024/digkult_emelt_gyakorlati_minta_-_melleklet.zip)  
(utoljára megtekintve: 2021. 11. 01.)
- J.M. Wing, Computational thinking. Communications ACM. 2006, 49, 33–35. (utoljára megtekintve: 2021. 11. 01.)
- 100/1997. (VI. 13.) Korm. rendelet az érettségi vizsga vizsgaszabályzatának kiadásáról. (1997)  
<https://net.jogtar.hu/jogszabaly?docid=99700100.KOR>. (utoljára megtekintve: 2021. 11. 01.)



14. M. Csernoch: Teaching word processing – the theory behind. *Teaching Mathematics and Computer Science*. (2009) 1, pp. 119–137.
15. M. Csernoch: Teaching word processing – the practice. *Teaching Mathematics and Computer Science*. (2010) (8)2, pp. 247–262.
16. M. Csernoch: Thinking Fast and Slow in Computer Problem Solving. *Journal of Software Engineering and Applications*. (2017) 10, 11-40.
17. SZTNH (2019) Mi a szellemi tulajdon-védelem célja? Módosítás dátuma: 2019. 08. 13.  
[https://www.sztnh.gov.hu/hu/mit-jelent/mi-a-szellemi-tulajdon-vedelem-celja?fbclid=IwAR0mQ4UkiYDEXkj8BDJHUT5G\\_mQebw6orLpLWtLQAiugtOgw4llwOZOuSg0](https://www.sztnh.gov.hu/hu/mit-jelent/mi-a-szellemi-tulajdon-vedelem-celja?fbclid=IwAR0mQ4UkiYDEXkj8BDJHUT5G_mQebw6orLpLWtLQAiugtOgw4llwOZOuSg0).  
(utoljára megtekintve: 2021. 11. 01.)
18. Oktatási hivatal: Kerettanterv az általános iskola 5–8. évfolyamára. *Vizuális kultúra*. (2020)  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/kerettanterv/Vizualis\\_kultura\\_F.docx](https://www.oktatas.hu/pub_bin/dload/kozoktatas/kerettanterv/Vizualis_kultura_F.docx). (utoljára megtekintve: 2021. 11. 01.)

## Forrás

19. [https://olympics.com/images/static/b2p-images/logo\\_color.svg](https://olympics.com/images/static/b2p-images/logo_color.svg). (utoljára megtekintve: 2021. 11. 01.)
20. D. Horváth: A kutya.  
<https://igazgyongyalapitvany.hu/download/Horvath-Dorina.pptx>. (utoljára megtekintve: 2021. 11. 01.)
21. *Megépült a világ legnagyobb vízerőműve*. Feltöltés dátuma: 2021. 07. 16. (2021)  
<https://alternativenergia.hu/megepult-a-vilag-legnagyobb-vizeromuve/94063>. (utoljára megtekintve: 2021. 11. 01.)
22. [https://upload.wikimedia.org/wikipedia/commons/thumb/c/ce/Ursa\\_Major\\_-\\_Ursa\\_Minor\\_-\\_Polaris.jpg/450px-Ursa\\_Major\\_-\\_Ursa\\_Minor\\_-\\_Polaris.jpg](https://upload.wikimedia.org/wikipedia/commons/thumb/c/ce/Ursa_Major_-_Ursa_Minor_-_Polaris.jpg/450px-Ursa_Major_-_Ursa_Minor_-_Polaris.jpg). (utoljára megtekintve: 2021. 11. 01.)



# Témák és ötletek az információkezelés tanítására

Holló Csaba

chollo@inf.u-szeged.hu  
SZTE TTIK Informatikai Intézet

**Absztrakt.** Az információs társadalom fejlődésével az információ egyre nagyobb értékkel bír, ami indokolja az információkezelés tanításának fontosságát. Ennek megfelelően az információkezelés különböző aspektusai a 2020-as NAT-ban is megjelennek. Jelen cikkben bemutatásra kerül néhány téma és ötlet az információkezelés tanítására.

**Kulcsszavak:** információs társadalom, információkezelés, információbiztonság.

## 1. Bevezetés

Viszonylag közhely az információkezelés fontossága, és annak szükségessége is, hogy ezt az tudást a diákok is megszerezzék ([1, 2, 3]). Az információkezelésről, biztonságról, beállítási és védekezési lehetőségekről, illetve azok tanításáról számos irodalom megjelent (például a [4] tananyagai, [5, 6]). Egy cikk nem alkalmas arra, hogy ilyen mennyiségű tartalom tanításának minden részletét tárgyalja, ezért bizonyos témákra egyáltalán nem, vagy csak felületesen térek ki. Az érintett témákban sem cél minden tárgyalandó ismeret felsorolása, csupán szeretnék a cikkben, illetve a mellékelt irodalmakon keresztül olyan tartalmakat megemlíteni, szempontokat, illetve gyakorlati ötleteket adni, amelyek az információkezeléshez tartozó témakörök tanítása során hasznosak lehetnek. A cikkben nem fogok kitérni arra sem, hogy a jelenlegi kerettanterveket figyelembe véve mely témákat, mikor, milyen óraszámokban lehetne tanítani, ugyanis ennek részletes elemzése egy külön cikket igényelne. Annnyit viszont fontosnak gondolok megemlíteni, hogy a kerettantervben leírtaknak megfelelően az információkezelés témaköréhez tartozó ismeretekre érdemes többször, azokat átismételve és fokozatosan bővítve tárgyalva visszatérni, annál is inkább, mert ennek a tudásnak számos elemére a diákoknak különböző korosztályokat átívelően (esetlegesen különböző mértékben) folyamatosan szükségük lehet, a képességek és normák kifejlődéséhez pedig fokozatos mélyítésre és gyakorlásra van szükség.

## 2. Szakmódszertani szempontok

Az információs technológiákhoz tartozó tanítandó ismeretek mennyisége az utóbbi időben jelentősen megnőtt, új kihívás elé állítva a tanárokat. Ugyanis, a hagyományos témakörök tanításakor többnyire elegendő volt arról meggyőződni, hogy a diákok megfelelően elsajátították a tananyagot, akkor valószínűsíthető volt, hogy szükség esetén a tanultakat használni is fogják. Az információs technológiához tartozó ismeretek esetén azonban, mivel a tanárnak nagyon kevés rálátása van a diákok online tevékenységére, el kellene érni, hogy a diákok valós helyzetekben a tanár hiányában is felismerjék, hogy hogyan kellene eljárni, és főleg hajlandóak legyenek azt megtenni akkor is, ha az olyan plusz munkával jár, ami nélkül a feladatot ugyan meg tudják oldani, csak kevésbé biztonságosan vagy etikusan. Ehhez pedig az kell, hogy kialakuljanak bennük alapvető biztonsági és etikai normák, és ily módon ők is fontosnak tartásák a tanultak betartását. Ez a tanuláshoz egy olyan módja, amikor az attitűd kialakítása legalább olyan fontos, mint a kompetenciák megszerzése, hiszen megfelelő attitűd hiányában a használatlan kompetenciák értéktelenek. Kérdés, hogy az attitűd kialakulását hogyan lehet elérni?

Természetesen ez is ott kezdődik, hogy a diáknak meg kell értenie, hogy milyen mechanizmusok játszódnak le, el kell sajátítania a szükséges ismereteket, és tudnia kell azokat alkalmazni.

Ugyanakkor nem mindegy, hogy ezeket az ismereteket hogyan dolgozzuk fel, mert „szárazon” elmondva nagyon hamar a feledés homályába merülhetnek. Igyekeznünk kell az ismereteket a diákok mindennapjaihoz kötni, érdekesen, és lehetőleg oly módon, hogy érzelmeket is kiváltunk belőlük, mert az a tudást maradandóbbá teszi ([7]), és segít az etikai normák kialakításában is. Ennek érdekében igyekszem a cikkekben, illetve a rendhagyó módon jelentős mennyiségben kevésbé tudományos cikkeket is tartalmazó irodalomjegyzékben, konkrét hétköznapi példákat, érdekes videók elérését is mellékelni.

Törley Gábor a [8] cikkében rámutat arra, hogy a biztonságtudatossággal kapcsolatos ismeretek tanításánál is igyekeznünk kell a Bloom-féle taxonómia minél magasabb szintjét elérni. Ez a többi információkezelési ismeret esetén is így van, és logikus összefüggésben van a meggyőzés sikerességével. Ugyanis, az elemzés, értékelés és teremtés szintjein egyre nagyobb esélyünk van a diákok meggyőzésére, és ily módon a biztonsági és etikai normák kialakítására.

A meggyőzés csak akkor elképzelhető, hogyha ismerjük és meg tudjuk cáfolni az esetlegesen a diákokat ebben gátló ellenérveket, amihez szükségeszerű, hogy a diákok kifejtsék a véleményüket, ütköztessék az érveiket, értékeljenek, és megoldásokon gondolkodjanak. Ehhez pedig nem elegendő „leadni” az anyagot, hanem beszélgetni kell a diákokkal, és olyan viszonyban kell lenni velük, illetve olyan légkört kell teremteni az osztályban és az órán, hogy merjék kimondani a gondolataikat.

### **3. Az információ megbízhatósága**

#### **3.1. Hitelesség és megbízhatóság**

Érdeemes közérthetően és praktikusán definiálnunk az információ hitelesség és megbízhatóság fogalmát. Az információ hiteles, ha a közzevetője megbízható és hitelesíti (felel érte) ([9]). Viszont érdemes figyelembe venni, hogy az általában megbízható közzevető is tévedhet. Ezért az információ megbízhatósága a hitelességnél erősebb elvárás, ami szükségessé teszi annak igazságtartalmának vizsgálatát több forrás, korábbi (tudományos) ismeretek, illetve racionális érvek alapján.

#### **3.2. Bizalom tényezők**

Hasznos megnézni a legfontosabb olyan tényezőket, amelyek növelik a bizalmat az olvasóban. Ilyenek lehetnek a design (profi megjelenés), az, hogy a tartalom az olvasó által már valószínűleg ismert információkra épít, érvelési hibáktól mentes, logikus, áttekinthető és átlátható felépítésű, a szerző elismert (megbízhatónak gondolt emberek bíznak már benne), a tartalom tárgyilagos (legalábbis látszólag elfogultságmentes), és alá van támasztva további szakirodalmakkal, referenciákkal. Az ilyen tényezőkről több okból is érdemes beszélni. Egyrészt azért, mert az információk közzétételénél fontos ezekre figyelni, ha azt szeretnénk, hogy mások megbízhatónak gondolják az általunk közzétett tartalmat. Másrészt olvasóként jó, ha ezek hiánya gyanút ébreszt bennünk. Harmadrészt pedig, a profi átverők használják is az ilyen tényezőket a bizalom megnyerésére, tehát csak ezek miatt még nem célszerű elhinni a tartalmat.

#### **3.3. A média hitelessége**

Bár ezt a témát a Vizuális kultúra, illetve a Mozgóképkultúra és médiaismeret tantárgyak is érintik, mégis érdemes néhány szót szólni arról, hogy hogyan működik az online média, hogyan lehet megélni vagy legalábbis pénzt szerezni (konkurencia, mennyiség a minőség rovására, szenzációhajhász és kattintásvadász oldalak, pl. [10]), a névtelenség és felelősség kérdéséről, és arról, hogy ezek hogyan fűggenek össze a valótlán információk megjelenésével és terjesztésével.

### 3.4. A közösségi oldalak szerepe

Sokan alapvetően a közösségi oldalakról, az azokban megjelenő megosztások alapján tájékozódnak, ezért fontos beszélni arról, hogy ez hogyan vezethet nem megfelelő ismeretszerzéshez.

Az ismerősök a híreket a saját „szemüvegükön” keresztül szűrik és módosíthatják is. Amint a [11, 12] cikkekben ismertetett tanulmányból kiderül, a felhasználók a kamuhíreket 70%-al nagyobb valószínűséggel osztják meg, általában a bennük levő több újdonság és élénk érzelmi reakciók miatt, ezért sokkal gyorsabban terjednek, mint az igaz hírek. Továbbá, bizonyos esetekben az álhírek melegágya lehet az is, hogy egyes vélekedések szerint valamilyen jellegű cenzúra miatt az igazságok nem kapnak napvilágot, és ezért a valótlanokat terjesztő ismerősök megosztásai hitelesebbeknek tűnnek.

A közösségi oldalakra másik jellemző jelenség a véleménybuborék kialakulása. A közösségi oldalakon az algoritmusok azon dolgoznak, hogy a felhasználó minél több időt töltsön az oldalon, ezért igyekeznek olyan megosztásokat beválogatni, amelyek valószínűleg tetszeni fognak a felhasználónak, vagy pedig (meglepő, esetlegesen szélsőséges tartalommal) felkeltik annak a figyelmét ([13]). Továbbá, a felhasználó is szűri az ismerőseit, többnyire azokkal barátkozik, kommunikál többet, és azoknak a megosztásait olvassa el, akikkel jelentős mértékben egyetért. Az ily módon kialakuló közösségek „nem látják egymást”, ezért hiányzik az egészséges vita is a tisztánlátáshoz. A hiányos tájékozódáson túl ez azért is probléma, mert annak elfogadása, hogy a nekem ellentmondó vitapartner nem ellenség, hanem egy másik szemszögből neki is lehet igaza, annak belátása, hogy ily módon a kulturált vita egy lehetőség a mélyebb megértéshez, illetve az ezeken alapuló kölcsönösen elfogadható megoldáskeresés képességének kialakítása, alapvető fontosságúak (lennének) a civilizált együttműködéshez, a problémák erőszakkal történő kezelésének elkerüléséhez.

### 3.5. Lehetőségek a megbízhatóság ellenőrzésére

Célszerű a közzétevő vizsgálatával kezdeni. Megbízhatók-e az eddig közzétett (megosztott) tartalmak: vannak-e ilyen tapasztalataink, vagy van-e ilyen jellegű információnk mások értékeléseiből, szerepel-e átverős oldalak listájában vagy jelez-e ilyen problémát az erre vonatkozó böngészőbővítőmenü? Feltételezhetően van-e megfelelő szaktudása a közzétevőnek, vagy a megosztott tartalom szerzőjének a tartalommal kapcsolatban? Lehet-e szándéka (érdeke) megtévesztésre?

Ezután célszerű megnézni, hogy elévült-e a tartalom? A legmaradandóbb a saját élmény, ezért érdemes a diákokkal olyan tartalomra kerestetni, amelyre az első találat tartalma már nem érvényes. Érdemes mutatni a tanulóknak olyan tartalmat is, amelyben félrevezetés céljából használnak olyan cikket, amely elavult vagy a tartalma könnyen félreértelmezhető, mint például az [14]-ben bemutatott 1976-os cikk a koronavírussal kapcsolatban.

A megbízhatóság eldöntésében fontos annak mérlegelése is, hogy a tartalom tény állít vagy vélemény fogalmaz meg, ezért érdemes ezt konkrét tartalmakkal gyakorolni. Tudatosítani kell a tanulóknak, hogy megfogalmazásaikban ők is figyeljenek a kettő közötti különbségre, mert a téves tényállításnak komolyabb jogi következményei is lehetnek.

Tipikus módszer az információ megbízhatóságának ellenőrzésére, hogy több forrást is megnézzünk. Itt viszont érdemes figyelembe venni, hogy a tartalmakat sok esetben a források is átvehetik egymástól (a komolyabb médiumok ezeket lehipotekozzák), tehát a többi forrás megbízhatóságát is értékelni kell, ugyanakkor hogyha hamis a tartalom, akkor van esélyünk olyan forrást is találni, amely cáfolja a valótlan vagy elferdített tartalmat.

Konkrét példákon keresztül be lehet mutatni a kamuhír-propaganda mögött rejlő taktikákat és jellegzetességeket, mint például érzelmekre ható tartalom, támadás valakik ellen, vagy más identitásának eltulajdonítása. Érdemes tudatosítani a tanulóknak, hogy az elkövetők egyes esetekben akár nyilvánvaló jogtalanságokat is bevállalnak, arra számítva, hogy a sértettnek nem éri meg a leplezésükhöz és jogi lépésekhez szükséges pénzt és energiát rááldozni, vagy ha mégis, az elkövetőknek

akkor is megérheti az esetleges jogi következményeket is megkockáztatni (például a reklámbevétel nagyobb lehet, mint a sértett esetleges hosszas pereskedésének következménye [15]). A tanulók a kamuhír taktikákat játékosan is gyakorolhatják (például a [16, 17, 18] cikkekben leírt módon, vagy az [alhirvadasz.hu](http://alhirvadasz.hu) oldalon) azért, hogy majd felhasználóként ezeket könnyebben felismerjék. Fontos továbbá a diákok figyelmét arra is felhívni, hogy a megfelelő tájékozódáshoz vegyék figyelembe a véleménybuborék jelenségét, illetve azt is, hogy a képek és videók is lehetnek manipulálva ([19]). Az ilyen manipulálások technikáiról és felismeréséről a diákok a Vizuális kultúra, illetve a Mozgóképkultúra és médiaismeret tantárgyakban is tanulnak, viszont legalábbis a mesterséges intelligencia ilyen jellegű alkalmazásairól a Digitális kultúra tantárgyban is célszerű lenne beszélni, érdekes történetek és videók felhasználásával, mint pl. a [20, 21, 22].

További ötleteket az álhírek felismerésére a [23, 24, 25., 26, 27, 28] cikkekben találunk.

A legfontosabb azonban, hogy az embernek igénye legyen a megbízható információkra, és hajlandó legyen az ehhez szükséges lépéseket megtenni, ugyanakkor valószínűleg ezt a legnehezebb elérni, ezért fontos arról is beszélgetni, hogy milyen következményei lehetnek a valótlan információk alapján történő döntéseknek és cselekedeteknek.

## 4. Az információ megosztása

### 4.1. Az információ értéke

A további tartalmak megalapozásaként érdemes megbeszélni néhány példát arra, hogy miért értékes az információ.

Ha nem jutunk egy információ birtokába, akkor eleshetünk lehetőségektől, tévesen ítélnetünk meg helyzeteket, illetve helytelen döntéseket hozhatunk.

Az információ manapság fizetőeszköz is ingyenesnek hirdetett szolgáltatásokhoz azáltal, hogy a regisztráció, illetve a szolgáltatások használata során adatokat osztunk meg magunkról, amelyeket a szolgáltatók eladhatnak olyanoknak, akik ezeket valamilyen célból fel szeretnék használni. Ugyanis, az információk, olyanok is, amikről nem is gondolnánk, felhasználhatók mások érdekében, vagy ellenük, például hasznos tudományos kutatásokra, de befolyásolásra, zsarolásra is. Az ártatlannak látszó információk felhasználására egy példa a [29] cikkben olvasható Chris "Birdman" Andersen esete, amikor az egyik áldozat életkorának ismerete kulcsfontosságú információ volt a zsaroláshoz, ezúttal pénzszerzés érdekében. A későbbi fejezetekben további példákat is látni fogunk az információk felhasználhatóságára.

### 4.2. Az információk gyűjtése

Érdemes beszélni azokról a lehetőségekről, hogy mások hogyan gyűjthetnek információt rólunk.

Nyilvánvaló, hogy amikor részt veszünk egy jelenléti eseményen, vagy nyilvános helyen cselekszünk valamit, akkor mások látnak, hallanak minket. Mivel az online térben kevésbé érzékeljük mások jelenlétét, ezért valamivel kevésbé tudatosulhat bennünk az, hogy online tevékenységeink során is ugyanez történik.

A fentieknél még kevésbé nyilvánvaló, hogy az internetes szolgáltatásokat nyújtó oldalak hogyan gyűjtik rólunk az információkat. Egyrészt, rengeteg adatot szolgáltatunk a nyilvános profiljainkban, különböző közösségi oldalakon, keresésekben, applikációkban megadott információkkal, lépésszámláló, naptár, helymeghatározás és más programok használatával. Érdekes feladat lehet a diákoknak az, hogy keresőben rákeressenek saját magukra, illetve lekérjék a róluk tárolt adatokat a Google Takeout szolgáltatásban, vagy a közösségi oldal fiókjukban. Másrészt, érdemes tisztázni, hogy a sütik ([30]), amelyek elvileg a kényelmünket szolgálják, további adatokat gyűjtenek be rólunk, amelyeket nem feltétlen csak annak az oldalnak szolgáltatnak, amelyet meglátogattunk, és amelyen ezeket en-

gedélyeztük, hanem más oldalaknak is, amelyeket az általunk meglátogatott oldal erre feljogosít, például azért cserébe, hogy azok szolgáltatásait használja. A Lightbeam böngészőkiegészítővel látványosan szemléltetni tudjuk, hogy egyes oldalak hogyan osztják meg egymással a rólunk gyűjtött adatokat ([31]), érdemes ezt a diákokkal kipróbálni.

Az információk gyűjtésének megbeszélése jó alkalom arra, hogy definiáljuk a személyes adatok és az adatvédelem fogalmait. A [32] definíciója alapján a személyes adat „bármely meghatározott, azonosított vagy azonosítható természetes személlyel [érintett] kapcsolatba hozható adat és az adatból levonható, az érintettre vonatkozó következtetés”, az adatvédelem pedig „a személyes adatok jogszerű kezelését, az érintett személyek védelmét biztosító alapelvek, szabályok, eljárások, adatkezelési eszközök és módszerek összessége”. Megjegyzendő, hogy a [32] definíciója szerint, a személyes adat az adatkezelés során mindaddig megőrzi e minőségét, amíg az adatkezelő rendelkezik azokkal a technikai feltételekkel, amelyek az adat és az érintett közötti kapcsolat helyreállításához szükségesek. Az adatvédelemmel kapcsolatos további kérdésekről a [33] irodalomban olvashatunk.

### 4.3. Megosztással kapcsolatos veszélyek

Az előzőekben leírt módon a cégek (különösen az olyan nagyok, mint a Google vagy Facebook) rengeteg adatot begyűjtenek. Érdemes tisztában lenni azzal is, hogy a látszólag személytelen adatok (általában ilyenek azok is, amiket a sütik gyűjtenek) egymással összekapcsolva, már személyes (azaz hozzánk köthető) adatokká válhatnak ([34]). Kimutatták például, hogy az amerikai lakosság 87%-át be lehet azonosítani kizárólag az irányítószámuk, nemük és születési dátumuk alapján ([35]). Ily módon az interneten is megjelenő tevékenységek (mit, mennyi ideig olvasunk, mire kattintunk, mit lájkolunk és mit nem, miket posztolunk, kereséseink, vásárlásaink, GPS koordinátáink), ismerőseink, törzsvásárlói kártyák, nyereményjátékok stb. elemzése révén az adatbrókerek nem csak felhasználói csoportokat képezhetnek, amelyeknek célzott üzeneteket (például reklámokat) lehet küldeni, de az információkat esetlegesen egészségünkre, vagy személyiségünkre vonatkozó személyes adatként is eladhatják, ami biztosítás, munkavállalás és más helyzetekben hátrányosan érinthet minket úgy, hogy nem is feltétlen lesz tudomásunk a történetekről ([36]). Persze, jogszabályok (például GDPR, illetve adatvédelmi törvények) korlátozzák a gyűjthető és tárolható adatokat és azok felhasználását ([33, 37]), de különösen felhasználói tudatosság hiányában ezek nem nyújtanak elegendő védelmet, továbbá általában sokkal könnyebb a problémákat megelőzni, mint kezelni. Ezért fontos tisztázni a személyes adatok fogalmát és az adatvédelmi tudnivalókat, amire az említett témakörök és példák jó lehetőséget nyújthatnak. Nyilvánvalóan manapság már nem lehet az online világból kiiratkozni, de azt megtehetjük, hogy legalább akkor, amikor erre lehetőségünk van, átgondoljuk, hogy milyen adatokat vagyunk hajlandóak megosztani magunkról, például érdemes-e részt vennünk egy nyereményjátékon, attól függően is, hogy milyen információkat kérnek cserébe.

Természetesen az adatainkat nem csak a nagy cégek vagy adatbrókerek gyűjthetik. A [38] videóban láthatunk példát arra, hogy hogyan lehet visszaélni egy nyereményjáték ürügyén megadott adatokkal. Ennek kapcsán is lehet beszélgetni arról, hogy miért érdemes különböző helyeken különböző jelszavakat használni, a kétfaktoros azonosításról, de a közzétett adataink nyilvánosságának korlátozásáról is. Utóbbi kapcsán érdemes megemlíteni, hogy a közzétett, önmagukban értelmetlen adataink is felhasználhatók arra, hogy csaló szándékkal nekünk küldött levélben elaltassák bennünk a gyanút, mivel annak olvasásakor azt gondolhatjuk, hogy ha valaki ennyi mindent tud rólunk, akkor biztosan ismer.

Talán kevésbé közismert veszélyként Gyurkó Szilvia hívta fel a figyelmet a [39] interjúban a következőkre. „Az #iamalone hashtag a legerősebb triggerje a pedofiloknak, mert pontosan tudják, hogy ott lesz a másik oldalon egy 13 éves kiskamasz, aki azt érzi, hogy nagyon egyedül van, senki nem szereti, baja van az élettel. Ha ráír, hogy milyen gyönyörű, vagy hogy „mutass többet magadból!”, akkor öt perc alatt rá tudja őt venni, hogy küldjön egy felvételt, amivel utána tudja zsarolni. És mivel egyedül van, senkitől nem fog tudni segítséget kérni.”. Fontos tehát a gyerekek figyelmét fel-

hívni arra, hogy a hashtagek is elárlhatnak róluk olyan információkat, melyek potenciális veszélyt jelenthetnek.

Hasonlóképpen szembesülhetnek a diákok a közösségi oldalakon megosztott tartalmak kellemetlen következményeivel a [40] videó megnézése kapcsán is, ami átvezet annak megbeszélésére, hogy hogyan élhetnek vissza akár az ismerősök is a megosztott információkkal és tartalmakkal. Tipikus példa erre, amikor általunk bizalmasan megosztott titkokat vagy képeket az ismerős az engedélyünk nélkül közzétesz vagy azok közzétételével zsarol. Ennek az egyik leggyakoribb példája, amikor a fiatalok körében meglehetősen gyakori szextingelésből ([41]), vagy korábbi intim viszonyból szerzett képeket tesznek közzé, akár pornográf oldalakon is telefonszámmal együtt. Az áldozat nevétségessé tétele sok esetben zaklatásba is átmehet, és az áldozatnak nehéz lelki terhet jelent, akár depressziót, vagy öngyilkossági gondolatokat is okozhat ([42]).

A [29] cikkben részletesen olvashatunk arról, hogy a rólunk megszerzett adatok birtokában hogyan lehet álprofil készíteni és azzal visszaélni. Ezek áttekintése mellett érdemes megbeszélni, hogy melyek azok az információk, amelyek felhasználhatók arra, hogy az álprofil valóságosnak tűnjön, és amelyekre ily módon figyelni kell, hogy lehetőség szerint minél kevesebb ilyen információ jusson illetéktelenek birtokába.

Tudatosítani kell a diákokban, hogy az előzőekben leírtak miatt is fontos az általuk használt szolgáltatások (első sorban a közösségi oldalak) beállításait úgy módosítani, hogy az adataikhoz való hozzáféréseket megfelelően korlátozzák.

#### 4.4. Szólásszabadság és felelősség

Az információ közlése kapcsán érdemes beszélni a szólásszabadság korlátairól. Arra, hogy hol a határ a véleménynyilvánítás és a hivatal vagy mások becsületének sértése között, nincs világszinten egységes szabályozás, például az Egyesült Államokban a szólásszabadságnak van nagyobb súlya, míg az EU-ban a személyiségi jogoknak ([43]), továbbá, ez a konkrét helyzettől is függ, bizonyos esetekben az érintett köteles többet eltűrni (például, ha közszereplő), de ez nem azt jelenti, hogy korlátlanul lehet másokat sértegetni. Különbség van a hamis tényállítás és a vélemény között is, például az [44] oldal éveken keresztül erre történő hivatkozással működhetett. Továbbá, vannak etikai szabályok is, amelyeket a közösségi létben be kell tartani akkor is, ha azt nem jogszabály írja elő, erre példaként érdemes megbeszélni a diákokkal, hogy nekik mennyire bántó lenne, hogyha például mások a hátuk mögött csúfolnák őket, tehát olyant ne tegyenek, aminek elszenvetői ők sem szeretnének lenni.

A felelősség témakörével már foglalkoztunk, a korábbiakban láttuk, hogy a megosztásokkal személyiségi jogokat vagy szerzői jogot sérthetünk. De azt a felelősséget is át kell gondolnunk, hogy az általunk megosztott információkkal másokat befolyásolunk, aminek súlyos következményei lehetnek. Aktualitásként lehet itt a koronavírussal kapcsolatos álhírekre is gondolni, de a fiatalokhoz talán közelebb álló példaként lehet a West-Balkán diszkótragédiáról ([45, 46]) is beszélni, ahol a pánik és az ebből következő tragédia alapja az volt, hogy valaki valótlánul kérelmet kiáltott, egyesek szerint a közösségi médiában is ezt osztották ([47]). Függetlenül attól, hogy a közösségi média megosztás igaz volt-e, az eset alkalmas annak megbeszélésére, hogy egy valótlán hír megosztásának hogyan lehetnek akár súlyos következményei. Érdemes tisztázni, hogy a felelősség akkor is fennáll, hogyha az adott információt nem mi osztjuk meg elsőként, hanem tovább osztjuk. Hogyha pedig egy (pl. Facebook) csoport adminisztrátorai vagyunk, akkor fokozott felelősségünk van abban, hogy hogyan kezeljük a csoport adatait, például a tagok beleegyezése nélkül egy eredetileg zárt csoportot nem tehetünk nyilvánossá még csak kizárólag kutatási célból vagy technikai okokból néhány percre sem. Ez a példa pedig átvezet annak tisztázására, hogy a kisebb nyilvánosság (például egy zárt Facebook csoportban való megosztás) csökkenti a cselekmény súlyát, de nem semmisíti meg azt. Konkrét példaként beszélhetünk egy romániai iskolában történt esetről ([48]), ahol Facebook csoportban tanárokat sértő szövegeket és képeket osztottak meg. Minden résztvevő büntetést kapott, de az adminisztrátor büntetése súlyosabb volt, mert ő felelt azért, ami a csoportban történt és nyilván tudott róla. Viszont



általában a kommentekért első sorban a szerző a felelős, az adminisztrátor akkor, ha tudomására jut, és nem távolítja el a jogtalan (például a másokat sértő) tartalmakat.

#### 4.5. A digitális lábnyom

A digitális lábnyom digitálisan tárolt személyes adat. A korábbi fejezetekben részletesen tárgyaltuk az általunk megosztott, illetve rólunk gyűjthető információkat, melyek mind a digitális lábnyom részei. Ebben a fejezetben ezt azzal egészíteném ki, hogy egyes szolgáltatásoknál (levelezésnél, közösségi oldalaknál) nyilatkozhatunk arról, hogy mi történjen az adatainkkal, a profilunkkal a halálunk után ([49]). Ez a diákoknak életkorukból kifolyólag többnyire inkább családtagjaik tekintetében lehet érdekes, ugyanakkor ennek kapcsán lehet beszélni az elhunyt személyek adatainak kezeléséről, illetve a mesterséges intelligencia alkalmazásáról is, a digitális lábnyomok alapján az elhunyt személy megte-remtéséről a digitális térben, illetve annak etikai és pszichológiai tényezőiről ([49, 50, 51, 52]).

#### 4.6. A megosztás előnyei

Az előző fejezetekben részletesen beszéltünk az információmegosztás veszélyeiről, de a diákok jogosan gondolnák elfogultnak a hozzáállásunkat, ha nem beszélnénk az előnyeiről is.

Nem nehéz olyan példákat találni, amikor valakinek előnye származott abból, hogy más által (legalábbis vele) megosztott információ birtokába jutott. De a megosztás előnyeihez tartozik a siker megosztásának lehetősége, illetve a közösségi értékelés is, mert ezek bizonyos esetekben motíválhatják a diákokot jobb minőségű munkára, nem feledkezve meg arról, hogy az viszont káros lehet, ha a diák az önértékelését kizárólag a visszajelzésektől teszi függővé. Hasonlóképpen, amint az előzőekben tárgyaltuk, veszélyeik mellett hasznosak is lehetnek a megosztott információk alapján nyújtott szolgáltatások, például az, ha a szolgáltatók a preferenciáinkról begyűjtött információk alapján ajánlanak további minket érdeklő tartalmakat.

Végül, az is nyilvánvaló, hogy az általunk készített és megosztott tartalmak érdekesek, értékesek, hasznosak lehetnek mások, illetve az utókor számára is, erre egy szokatlan példa a [53] emlékdal, mely egyben kordokumentum is.

#### 4.7. Megosztás előtt

Ajánlott átgondolni, hogy az adott tartalmat miért és kikkel szeretnénk megosztani, és a megosztást annak megfelelően korlátozni.

Érdeemes felhívni a diákok figyelmét arra, hogy csak olyasmit érdemes közzétenni, amiben biztosak vagyunk, hogy később sem fogjuk megbánni, mert nem biztos, hogy később lehetőségünk lesz törölni, hiszen más személyek vagy oldalak is készíthetnek arról másolatot. Gondolni kell arra, hogy mi is változunk, a környezetünk is változik, és ami ma jónak tűnik, azt egy idő után vállalhatatlannak érezhetjük, vagy hátrányosan érinthet. Továbbá, ha tudjuk is törölni, ha valakit megbántottunk vele, azt nem lehet meg nem történtté tenni.

Ajánlott átgondolni, hogy a megosztásból a többiek mit tudhatnak meg rólunk, és az felhasználható lehet-e ellenünk.

Figyelni kell arra is, hogy megosztásainkkal nem csak magunknak, hanem másoknak is árthatunk. Lehet, hogy az információ, kép, videó tulajdonosa, vagy a rajta szereplő személyek, valamilyen okból nem szeretnék, hogy azt megosszuk, és a megosztással árthatunk is anélkül, hogy ez szándékunkban állna. Érdemes a diákokkal megbeszélni, hogy voltak-e ilyen jellegű élményeik.

Természetesen ki kell térni a megosztás adatvédelmi és szerzői jogi vonatkozásaira is, hiszen a másokkal kapcsolatos információk megosztása személyes adatok kezelésének minősül, és ily módon a megosztáshoz is ki kell kérnünk az érintettek beleegyezését.

Érdeemes a diákokkal megismertetni azon információkezeléssel kapcsolatos jogszabályi részleteket és büntetési tételeket, amelyekben elkövetőként vagy sértettként ők is érintettek lehetnek.

## 5. Adatbiztonság

A [32] definíciója szerint az adatbiztonság „az adatok jogosulatlan megszerzése, módosítása és megsemmisítése elleni műszaki és szervezési megoldások rendszere.”

Ebben a témakörben célszerű beszélni a lehetséges támadásokról és kockázatokról (vírusok, férgek, trójai programok, kémprogramok, láncclevél, spam, adathalászat, biztonsági rések stb.), illetve arról, hogy ilyen támadások célpontjai olyan más szolgáltatók is lehetnek, ahol a mi jelszavainkat tárolják, és ezért ajánlott, hogy különböző oldalakon különböző jelszavakat használjunk, ráadásul ezek nehezen feltörhetőek legyenek ([54, 55]) és azokat rendszeresen változtassuk meg. Sok erős jelszót megjegyezni viszont csak akkor lehet, ha ezeket valamilyen algoritmus szerint találjuk ki (például [56]), segíthet a kétfaktoros azonosítás ([57, 58]) használata, melynek azonban kockázatai is vannak ([59, 60]), illetve használhatunk jelszókezelő programot is. További lehetőség a biometrikus azonosítás ([61]), ami sokféleképpen azonosíthat (például ujjlenyomat, retina, hang, DNS, arc, de akár gépelés, járás, mozgás alapján is), és aminek a digitálisan megváltoztatott verzióját célszerű tárolni azért, hogy ellopás esetén megváltoztatható legyen. Érdeemes megmutatni, hogy vannak olyan oldalak, ahol becslést kaphatunk arra, hogy mennyi idő alatt törhető fel egy adott jelszó (például <https://www.security.org/how-secure-is-my-password/>), ahol meg lehet nézni, hogy adott jelszó szerepel-e az általuk gyűjtött feltört jelszavak adatbázisában (például [haveibeenpwned.com/](https://haveibeenpwned.com/)), illetve ahol segítséget kaphatunk zsarolóprogramok ellen (pl. <https://www.nomoreransom.org/hu/>).

Az adatbiztonság témakörében számos további internetes szolgáltatás (például elektronikus ügyintézés, kereskedelem, bankolás stb.) biztonságára is ki kell térni, melyek részletezése azonban meghaladná ezen cikk kereteit, de jónéhány tartalmi ötletet kaphatunk a [62] videóban, amelyek megbeszélését érdemes kiegészíteni tényleges történetekkel, illetve a diákok tapasztalataival, mert a megbeszéltek akkor lesznek maradandóak, hogyha konkrét példákhoz kapcsolódnak.

## 6. Szerzői jogok

Különösen a mai fiatalok körében a megosztás természetes és gyakori jelenség, és sok esetben nem gondolják át ennek hátterét, nem csak biztonság, másoknak okozott esetleges kár, hanem szerzői jogi szempontból sem, ezért is fontos erről beszélni.

A kerettantervben a szerzői jog explicit módon nem szerepel, csak a hivatkozás szabályai, jogszabályok, és az etikus információkezelés ([63, 64]), amibe viszont véleményem szerint a szerzői jog több megközelítéssel is beletartozik. Egyrészt, a szerző és a szerzői jog meghatározása szoros kapcsolatban áll egymás munkájának a tiszteletével, függetlenül attól, hogy azt mennyire gondoljuk értékesnek, továbbá a hivatkozások is csak ennek kontextusában nyernek értelmet. Dolgozatainkban a diákok is felhasználják mások munkáit, ezért beszélnünk kell a felhasználás kategóriáiról, különös tekintettel a szabad felhasználás oktatási lehetőségeire és korlátaira ([65]), és nem csak általánosságban, hanem konkrét eseteket megbeszélve, mint például az Európai Unió Bíróságának ítélete az iskolai dolgozatban használt és közzétett fotó kapcsán ([66]), vagy a Jerusalema kihívással kapcsolatos szerzői jogok ([67]). Természetesen a megoldást is meg kell mutatunk, azaz célszerű ismertetni olyan oldalakat is, ahonnan zenéket, képeket, illetve videókat ingyenes le lehet tölteni (mint például a [68, 69] cikkekben ismertettek, továbbá [ncs.io](https://ncs.io), [free-stock-music.com](https://free-stock-music.com), [danosongs.com](https://danosongs.com), [pixabay.com](https://pixabay.com), [depositphotos.com](https://depositphotos.com), [unsplash.com](https://unsplash.com), [pexels.com](https://pexels.com), [foter.com](https://foter.com), [gratisography.com](https://gratisography.com) stb.).

Beszélnünk kell az internetes szolgáltatások felhasználási feltételeiről, a számukra átruházott jogkörökről, továbbá a programok licenelési kategóriáiról, a letölthető digitális termékek használatáról és megosztásáról is ([70]), lehetőleg a diákok által is használt programokat és tartalmakat használva példaként.

## 7. Összefoglalás

Az információs társadalom fejlődésének következtében az információkezelés témakörében a diákokkal jelentős mennyiségű ismeretet szükséges megbeszelnünk, melyeknek jelen cikkben csak egy részét érintettük, különböző mélységekben, első sorban felvillantva bizonyos ismereteket, összefüggéseket és ezekhez köthető mindennapi példákat. Remélhetőleg ezek a tartalmak elgondolkodtatják a diákokat, segítik az egyes témák megértését elmélyíteni, az ismeretek megbeszélését a Bloom-féle taxonómia minél magasabb szintjére emelni, és ily módon a kívánt biztonsági és etikai normákat jobban kialakítani. Lehetőség szerint ezek kiegészíthetők a [8]-ban is említett más eszközökkel (virtuális valóság, szimuláció, társas- és szituációs játékok, színházi nevelés, pszichodráma stb.), illetve továbblépésként ilyenek kifejlesztésével. Végül pedig, más megközelítésből visszatérve a Bloom-féle taxonómiára, azért is fontos, hogy a diákok képesek legyenek az ismereteket megfelelő minőségben érteni, használni, értékelni, és megoldásokat kidolgozni a felmerülő problémákra, mert a fejlődő információs technológia kihívásainak a jelenlegi diákok általi kezelése a jövőben jelentősen befolyásolhatja az életminőséget.

## Irodalom

1. Dzindzisz Sztéfan: *Iskolákban tanítaná a kormány, mi a fake news*, 2018. január 15., [https://www.napi.hu/tech/iskolakban-tanitana-a-kormany-mi-a-fake-news.654724.html?utm\\_source=index.hu&utm\\_medium=doboz&utm\\_campaign=link&fbclid=IwAR2i\\_lyETfx-hz96M0TEfQADN1aQUS01iZYqLcOa6RWOvFvNp\\_-JPDx-gA&](https://www.napi.hu/tech/iskolakban-tanitana-a-kormany-mi-a-fake-news.654724.html?utm_source=index.hu&utm_medium=doboz&utm_campaign=link&fbclid=IwAR2i_lyETfx-hz96M0TEfQADN1aQUS01iZYqLcOa6RWOvFvNp_-JPDx-gA&) (utoljára megtekintve: 2021.11.09.)
2. Political Capital: *Álhírek elleni küzdelem az oktatásban*, 2019.11.28., [https://politicalcapital.hu/hireink.php?article\\_read=1&article\\_id=2476](https://politicalcapital.hu/hireink.php?article_read=1&article_id=2476) (utoljára megtekintve: 2021.11.09.)
3. Bálint Kata, Hunyadi Bulcsú, Krekó Péter, László Róbert, Molnár Csaba: *Álhírek elleni küzdelem az oktatásban*, Budapest, 2019. november, [https://politicalcapital.hu/pc-admin/source/documents/pc\\_alhitek\\_elleni\\_kuzdelem\\_az\\_oktatasban\\_20191128.pdf](https://politicalcapital.hu/pc-admin/source/documents/pc_alhitek_elleni_kuzdelem_az_oktatasban_20191128.pdf) (utoljára megtekintve: 2021.11.09.)
4. NJSZT: *Tananyagok*, [https://njszt.hu/hu/form/tananyag-letoltese?tananyag=konyv\\_it\\_biztonsag\\_2019](https://njszt.hu/hu/form/tananyag-letoltese?tananyag=konyv_it_biztonsag_2019) (utoljára megtekintve: 2021.11.09.)
5. Dr. Abonyi-Tóth Andor, Dr. Turcsányi-Szabó Márta: *A digitális írástudás fejlesztésének lehetőségei*, <https://dl-sulinet.educatio.hu/download/letoltheto-dokumentumok/Digitalis-irastudas.pdf> (utoljára megtekintve: 2021.11.09.)
6. Európa Pont: *Harc az álhírek ellen*, [https://europapont.blog.hu/2018/03/13/harc\\_az\\_alhitek\\_ellen](https://europapont.blog.hu/2018/03/13/harc_az_alhitek_ellen) (utoljára megtekintve: 2021.11.03.)
7. Freund Tamás: *Agyhullámok, tanulás, kreativitás*, In: Pedagógiai Esték, Szeged, 2016. 11. 22., <https://www.youtube.com/watch?v=J13f5so7y2c> (utoljára megtekintve: 2021.11.09.)
8. Törley Gábor: *A biztonságtudatosság oktatási módszerei: a bevonódás szerepe*, In: „Közigazgatás Napja Konferencia 2021” című tanulmánykötetben megjelent tanulmány, 2021.
9. Prusi: *Szakdolgozat az információkezelésről*, 2017. 09. 02., [https://prusi.blog.hu/2017/09/02/szakdolgozat\\_az\\_informaciok\\_hitelessegerol\\_i](https://prusi.blog.hu/2017/09/02/szakdolgozat_az_informaciok_hitelessegerol_i) (utoljára megtekintve: 2021.11.03.)
10. Hanula Zsolt: *Majdnem megtanultam három nyelvet egy hónap alatt*, <https://index.hu/tech/2016/08/29/majdnem-megtanultam-harom-nyelvet-egy-honap-alatt/> (utoljára megtekintve: 2021.11.06.)

11. *Fekétén-fehéren kiderült: szívesen osztunk meg kamuhíreket.* In: HVG, 2018.03.18.  
[https://hvg.hu/tudomany/20180318\\_kamuhirek\\_terjedese\\_a\\_twitteren\\_mit\\_kutatas](https://hvg.hu/tudomany/20180318_kamuhirek_terjedese_a_twitteren_mit_kutatas) (utoljára megtekintve: 2021.11.06.)
12. Soroush Vosoughi, Deb Roy, Sinan Aral: *The spread of true and false news online.* In: SCIENCE, 2018.03.09., Vol 359, Issue 6380, pp. 1146-1151, DOI: 10.1126/science.aap9559,  
<https://www.science.org/doi/10.1126/science.aap9559> (utoljára megtekintve: 2021.11.06.)
13. Lovász Dávid: *A véleménybuborék jelensége a közösségi médiában.* In: Kalauz, PTE Könyvtár és Tudásközpont, 2017. 12. 15.,  
<https://kalauz.lib.pte.hu/velemenybuborek-jelensege-kozossegi-mediaban/> (utoljára megtekintve: 2021.11.06.)
14. Marinov Iván: *Koronavírus: mit bizonyít a neten terjedő 1976-os cikk?* 2020. 08. 17.,  
<https://www.urbanlegends.hu/2020/08/koronavirus-mit-bizonyit-a-neten-terjedo-1976-os-cikk/> (utoljára megtekintve: 2021.11.03.)
15. *Együtt nyomoznak és perelnek Liptaiék: milliós család áldozatai lettek.* Bors, 2017.11.21.,  
<https://www.borsonline.hu/celeb/egyutt-nyomoznak-es-perelnek-liptai-ek-millios-csalad-aldozatai-lettek/143306> (utoljára megtekintve: 2021.11.06.)
16. Mák Gabi: *Gyárts te is kamu híreket!*  
[https://divany.hu/eletem/2018/02/22/gyarts\\_te\\_is\\_kamu\\_hireket/](https://divany.hu/eletem/2018/02/22/gyarts_te_is_kamu_hireket/) (utoljára megtekintve: 2021.11.06.)
17. Marinov Iván: *Hogy állsz a tényekkel? – Igaz/Hamis kvíz gyerekeknek.*  
[https://www.urbanlegends.hu/2021/10/annamari-bon-fake-konyv-kviz/?fbclid=IwAR3X\\_s2ICSIYxE9vQanOB-jSQsGx4Rt3sDzsRah8mXn8FCT0egQWIKHw6ZA](https://www.urbanlegends.hu/2021/10/annamari-bon-fake-konyv-kviz/?fbclid=IwAR3X_s2ICSIYxE9vQanOB-jSQsGx4Rt3sDzsRah8mXn8FCT0egQWIKHw6ZA) (utoljára megtekintve: 2021.11.06.)
18. *Ezt a vizsgát kötelezővé tennék az érettségien: pofonegyszerű játékkal kell felismerni a kamuhíreket.* In: HVG, 2017. 07. 18.,  
[https://hvg.hu/tudomany/20170718\\_factitious\\_kamu\\_hirek\\_hoax\\_atveros\\_oldalal\\_alhitek\\_jatek](https://hvg.hu/tudomany/20170718_factitious_kamu_hirek_hoax_atveros_oldalal_alhitek_jatek) (utoljára megtekintve: 2021.11.06.)
19. *Így verhet át a média!* In: 6 lépés  
[https://youtu.be/w\\_HY4dliON4](https://youtu.be/w_HY4dliON4) (utoljára megtekintve: 2021.11.06.)
20. *Erre is képes a mesterséges intelligencia!* 2019.12.13.,  
<https://youtu.be/tkimwiVzF5c> (utoljára megtekintve: 2021.11.09.)
21. *Nem létező személyek készítése.* In: PosztmodeM, 2019. máj. 24.,  
[https://youtu.be/UyiTeV\\_B81c](https://youtu.be/UyiTeV_B81c) (utoljára megtekintve: 2021.11.09.)
22. Molnár Csaba: *Először csaltak ki pénzt deepfake hanggal, és rögtön van magyar szál.* 2019.09.05.,  
[https://index.hu/techtud/2019/09/05/eloszor\\_csaltak\\_ki\\_penzet\\_deepfake\\_hanggal\\_es\\_rogton\\_van\\_magyar\\_szal/?fbclid=IwAR2kSPHfMIEzTNf9c2HmyfBymqUlyEWxt\\_j1Se4WQTFnjvLUJaNX9WIVrw](https://index.hu/techtud/2019/09/05/eloszor_csaltak_ki_penzet_deepfake_hanggal_es_rogton_van_magyar_szal/?fbclid=IwAR2kSPHfMIEzTNf9c2HmyfBymqUlyEWxt_j1Se4WQTFnjvLUJaNX9WIVrw) (utoljára megtekintve: 2021.11.09.)
23. Fülöp Hajnalka: *Álbíre.* In: GimiMore Plus, 2019.11.07.,  
[https://youtu.be/wxo\\_ib4WqEM?list=PLKd2de-K0pxnMNH90DtNeAEfQs7ralw6](https://youtu.be/wxo_ib4WqEM?list=PLKd2de-K0pxnMNH90DtNeAEfQs7ralw6) (utoljára megtekintve: 2021.11.06.)
24. Marinov Iván: *Kamuhírek az iskolában 1. rész: Hogyan tanítsuk a kamuhírek felismerését az iskolában?* 2017.06.16.,  
<https://www.urbanlegends.hu/2017/06/kamuhirek-iskolai-oktatasa/> (utoljára megtekintve: 2021.11.06.)
25. Marinov Iván: *Kamuhírek az iskolában 2. rész: a kritikus gondolkodás oktatása.* 2017.08.30.,  
<https://www.urbanlegends.hu/2017/08/kamuhirek-az-iskolaban-2-resz-a-kritikus-gondolkodas-oktatasa/?fbclid=IwAR0soMPfTvS8PLA49blerCcapZMugruOvfUSBpe0F5ms9Rvs9Gesfq7WMU4> (utoljára megtekintve: 2021.11.06.)
26. Marinov Iván: *Kamuhírek az iskolában 3. rész: óravázlatok tanároknak.* 2017.10.20.,  
[https://www.urbanlegends.hu/2017/10/kamuhirek-az-iskolaban-3-resz-oravazlatok-tanaroknak/?fbclid=IwAR3EydxTwl\\_KJ4gQKjPQz4-frudrIMxMgqzrv56zhMozuoOH60w79A-foE](https://www.urbanlegends.hu/2017/10/kamuhirek-az-iskolaban-3-resz-oravazlatok-tanaroknak/?fbclid=IwAR3EydxTwl_KJ4gQKjPQz4-frudrIMxMgqzrv56zhMozuoOH60w79A-foE) (utoljára megtekintve: 2021.11.06.)

27. Marinov Iván: *Kamuhírek az iskolában 4.: ime egy magyar nyelvű program!* 2017.11.08., [https://www.urbanlegends.hu/2017/11/kamuhirek-az-iskolaban-4-ime-egy-magyar-nyelvu-program/?fbclid=IwAR0mweK-vJvL9sjy9waINWVnOvL7fv6v8\\_L0Zz4P\\_Wh1B-5tMPeeIempxrg](https://www.urbanlegends.hu/2017/11/kamuhirek-az-iskolaban-4-ime-egy-magyar-nyelvu-program/?fbclid=IwAR0mweK-vJvL9sjy9waINWVnOvL7fv6v8_L0Zz4P_Wh1B-5tMPeeIempxrg) (utoljára megtekintve: 2021.11.06.)
28. Laza Bálint, Pintér Dániel Gergő: *Így ismerd fel, és különböztess meg a valódi szakértőt az általszakértőtől!* 2019.08.26., [https://media20.blog.hu/2019/08/26/igy\\_ismerd\\_fel\\_es\\_kulonboztess\\_meg\\_a\\_valodi\\_szakertot\\_az\\_alszakertotol?fbclid=IwAR2MjHCOIdAgJqWi6gLaN\\_WXtkbAuHYVBxS6iL2e\\_DWOrbtvJV02sszhyQc](https://media20.blog.hu/2019/08/26/igy_ismerd_fel_es_kulonboztess_meg_a_valodi_szakertot_az_alszakertotol?fbclid=IwAR2MjHCOIdAgJqWi6gLaN_WXtkbAuHYVBxS6iL2e_DWOrbtvJV02sszhyQc) (utoljára megtekintve: 2021.11.06.)
29. Holló Csaba: *Álprofilok használata az etikus és biztonságos internethasználat tanításában.* In: INFODIDACT 2018 (2018.11.22-24), Informatika Szakmódszertani Konferencia, elektronikus kiadványa, 59-70, Zamárdi, Hungary, április, 2019, ISBN: 978-615-80608-2-0., <http://konferencia.inf.elte.hu/infodidact/InfoDidact18/Manuscripts/HCs.pdf> (utoljára megtekintve: 2021.11.03.)
30. Nemzeti Kibervédelmi Intézet: *Tájékoztató a sütikről.* <https://nki.gov.hu/intezet/tartalom/tajekoztato-a-sutikrol/> (utoljára megtekintve: 2021.11.07.)
31. Kulcsár Bálint: *A digitális lábnyomaink,* 2016.03.03., <https://www.adatvedelmirendelet.hu/internet/a-digitalis-labnyomaink/>, (utoljára megtekintve: 2021.11.07.)
32. *Adatvédelmi értelmező szótár.* <https://www.naih.hu/adatvedelmi-szotar> (utoljára megtekintve: 2021.11.14.)
33. Alexin Zoltán: *A személyes adatok védelmének jogi, etikai és informatikai kérdései.* Typotex, 2011, <http://publicatio.bibl.u-szeged.hu/1583/> (utoljára megtekintve: 2021.11.09.)
34. Nagy Attila Károly: *Nevesíthető a felhasználó az anonim böngészési adatokból is.* In: Index, 2017.08.04., [https://index.hu/tech/2017/08/04/nevesitheto\\_a\\_felhasznalo\\_az\\_anonim\\_bongeszesi\\_adatokbol\\_is/?fbclid=IwAR2oIfzp64tPDRICSczUlseJnayN0jrMutJ4wTheLE8CxdWtjktXvNq9ro4](https://index.hu/tech/2017/08/04/nevesitheto_a_felhasznalo_az_anonim_bongeszesi_adatokbol_is/?fbclid=IwAR2oIfzp64tPDRICSczUlseJnayN0jrMutJ4wTheLE8CxdWtjktXvNq9ro4) (utoljára megtekintve: 2021.11.07.)
35. L. Sweeney: *Simple Demographics Often Identify People Uniquely.* Carnegie Mellon University, Data Privacy Working Paper 3. Pittsburgh 2000., <https://dataprivacylab.org/projects/identifiability/index.html> (utoljára megtekintve: 2021.11.07.)
36. Madhumita Murgia: *Hogyan adták el adatbrókerek a személyiséget.* In: TED, 2017.04., [https://www.ted.com/talks/madhumita\\_murgia\\_how\\_data\\_brokers\\_sell\\_your\\_identity?language=hu](https://www.ted.com/talks/madhumita_murgia_how_data_brokers_sell_your_identity?language=hu) (utoljára megtekintve: 2021.11.07.)
37. *Európai adatvédelmi jogi kézikönyv.* 2018, <https://op.europa.eu/en/publication-detail/-/publication/5b0cfa83-63f3-11e8-ab9c-01aa75ed71a1/language-hu/format-PDF/source-111961710> (utoljára megtekintve: 2021.11.09.)
38. Digitális Kutatások Intézete: *Megdöbbentő adatkísérlet.*, 2018.10.03., <https://www.facebook.com/digitaliskutatasok/videos/140021440286597/> (utoljára megtekintve: 2021.11.08.)
39. Janecska Kata: *A biztonságérzetünket zúzta szét a Kaleta-ügy. Interjú Gyurkó Szilviával, az UNICEF korábbi gyermekjogi igazgatójával.* In: Index, 2020.07.10., [https://index.hu/belfold/2020/07/10/kaleta\\_gabor\\_gyermekporno\\_szabalyozas\\_itelkezesei\\_gyakorlat\\_gyurko\\_szilvia\\_interju/](https://index.hu/belfold/2020/07/10/kaleta_gabor_gyermekporno_szabalyozas_itelkezesei_gyakorlat_gyurko_szilvia_interju/) (utoljára megtekintve: 2021.11.08.)
40. *A posztolások veszélyei.* In: Kék Vonal Komputer Klubház Kecskemét, 2014.10.15., <https://youtu.be/5ha0E-oJrOw> (utoljára megtekintve: 2021.11.08.)
41. *Szexting.* In: Gyerek a neten, <https://gyerekaneten.hu/szocikk/Szexting> (utoljára megtekintve: 2021.11.08.)
42. Grünfelder Borbála, Holló Csaba: *Az informatikatanár lehetőségei az internetes zaklatás megelőzésében és kezelésében.* In: INFODIDACT 2019 (2019. 11. 21-23), Informatika Szakmódszertani Konferencia, elektronikus kiadványa, 73-86, Zamárdi, Hungary, január, 2020, ISBN: 978-615-80608-3-7., <https://people.inf.elte.hu/szlavi/InfoDidact19/Manuscripts/GBHCs.pdf> (utoljára megtekintve: 2021.11.03.)

43. Horváth Bence: *Az internetes gyűlöletbeszéd nem tartozik a szólásszabadság hatálya alá az Emberi Jogok Európai Bírósága ítélete szerint.* 2017.07.20.,  
<https://444.hu/2017/07/20/az-internetes-gyuloletbeszed-nem-tartozik-a-szolasszabadsag-hatalya-ala-az-emberi-jogok-europai-birosaga-itelete-szerint> (utoljára megtekintve: 2021.11.10.)
44. MarkMyProfessor, markmyprofessor.com (utoljára megtekintve: 2021.11.10., a megtekintés pillanatában nem elérhető)
45. *Tragédia a West-Balkán szórakozóbelyen.*  
[https://hu.wikipedia.org/wiki/Trag%C3%A9dia\\_a\\_West-Balk%C3%A1n\\_sz%C3%B3rako%C3%B3z%C3%B3helyen](https://hu.wikipedia.org/wiki/Trag%C3%A9dia_a_West-Balk%C3%A1n_sz%C3%B3rako%C3%B3z%C3%B3helyen) (utoljára megtekintve: 2021.11.08.)
46. *Emlékezzünk a West Balkán áldozataira Facebook csoport.*  
<https://www.facebook.com/tragediawestbalkan/> (utoljára megtekintve: 2021.11.08.)
47. *Diszkréttragédia: a közösségi média leszerepelt?* 2011.01.16.,  
[https://hvg.hu/itthon/20110116\\_kozossegi\\_media\\_diszkréttragedia](https://hvg.hu/itthon/20110116_kozossegi_media_diszkréttragedia) (utoljára megtekintve: 2021.11.08.)
48. *Előfordulhatna-e egy Sincsi-ügy a kolozsvári magyar középiskolákban?* In: Transindex, 2015.03.05.,  
<https://eletmod.transindex.ro/?cikk=24990> (utoljára megtekintve: 2021.11.09.)
49. *Hogyan kezelhetjük elhunyt szeretteink online hagyatékát?* Nemzeti Média- és Hírközlési Hatóság, 2019.08.13.,  
[https://nmhh.hu/cikk/206320/Hogyan\\_kezelhetjuk\\_elhunyt\\_szeretteink\\_online\\_hagyatekat?](https://nmhh.hu/cikk/206320/Hogyan_kezelhetjuk_elhunyt_szeretteink_online_hagyatekat?) (utoljára megtekintve: 2021.11.09.)
50. Schleer Tamás: *Digitális illúzióink.* In: TEDxYouth@Budapest, 2016.10.27.,  
<https://youtu.be/biBOt61uesE?list=PLDlsemvZx1fjywh1gXT8-tPEPy3B1qgKY> (utoljára megtekintve: 2021.11.09.)
51. *Megteremtené az elhunyt családtaggal való időöltés lehetőségét egy magyar alapítású cég.* HVG, 2020. június. 11.,  
[https://hvg.hu/tudomany/20200611\\_realic\\_hybri\\_recall\\_funkcio\\_kiterjesztett\\_valosag\\_virtualis\\_valosag\\_a\\_r\\_vr](https://hvg.hu/tudomany/20200611_realic_hybri_recall_funkcio_kiterjesztett_valosag_virtualis_valosag_a_r_vr) (utoljára megtekintve: 2021.11.09.)
52. Bodnár Zsolt: *Virtuális valóságban találkozott újra halott lányával egy koreai nő.*  
<https://qubit.hu/2020/02/09/virtualis-valosagban-talalkozott-ujra-halott-lanyaval-egy-koreai-no> (utoljára megtekintve: 2021.11.09.)
53. Karapándzsity Kristóf: *Sors-szinkópa.*  
<https://www.hashtagmagazin.net/elmerengo/sors-szinkopa> (utoljára megtekintve: 2021.11.09.)
54. Dr. Kulcsár Zoltán, Útmutató *biztonságos jelszókezeléshez.* 2019.03.22.,  
<https://www.xn--adatvdelem-f7a.hu/2019/03/utmutato-biztonsagos-jelszokezeleshez/> (utoljára megtekintve: 2021.11.09.)
55. Major Szabolcs: *Másodpercenként 100 milliárd jelszót fejt meg egy gép: a tiéd sincs biztonságban.*  
<https://computerworld.hu/biztonsag/masodpercenkent-100-milliard-jelszot-fejt-meg-egy-gep-a-tied-sincs-biztonsagban-284833.html> (utoljára megtekintve: 2021.11.09.)
56. *Hogyan készíts biztonságos jelszavakat?* 2019. 09. 25.,  
<https://hellobiznisz.hu/hogyan-keszits-biztonsagos-jelszavakat/> (utoljára megtekintve: 2021.11.09.)
57. *Védekezz duplán: ez az 5 legjobb app a fiókjaid védelmére.* 2020 01.02.,  
<https://pcworld.hu/pcwpro/ketfaktoros-belepes-autentikator-appok-tipp-272545.html> (utoljára megtekintve: 2021.11.09.)
58. *Mi az a kétfaktoros azonosítás és hogyan tudom beállítani?* Nexus Learning,  
<https://support.nexuslearning.com/kb/00023> (utoljára megtekintve: 2021.11.09.)
59. Major Szabolcs: *A Microsoft szerint nem biztonságos a mobilos kétfaktoros azonosítás.* 2020.11.17.,  
<https://computerworld.hu/biztonsag/a-microsoft-szerint-nem-biztonsagos-a-mobilos-ketfaktoros-azonositas-287022.html> (utoljára megtekintve: 2021.11.09.)
60. Fekete Gábor: *SIMultán átverés: figyelj oda a kártyádra!* 2021.02.04.,  
<https://pont-most.hu/gep/simultan-atveres-figyelj-oda-a-kartyadra/> (utoljára megtekintve: 2021.11.09.)
61. *Alapvető információk a biometrikus azonosítás témájáról.* Biometrikus,  
<https://biometrikus.hu/> (utoljára megtekintve: 2021.11.10.)

62. Magyarósi Csaba: *Hogy NE lopják el ONLINE a pénzed?* Karantén Kisokos, <https://youtu.be/aEykr1mjow> (utoljára megtekintve: 2021.11.09.)
63. *Kerettanterv Digitális kultúra 5–8. évfolyam.*  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/kerettanterv/Digitalis\\_kultura\\_F.docx](https://www.oktatas.hu/pub_bin/dload/kozoktatas/kerettanterv/Digitalis_kultura_F.docx) (utoljára megtekintve: 2021.11.09.)
64. *Kerettanterv Digitális kultúra 9–11. évfolyam.*  
[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/kerettanterv/Digitalis\\_kultura\\_K.docx](https://www.oktatas.hu/pub_bin/dload/kozoktatas/kerettanterv/Digitalis_kultura_K.docx) (utoljára megtekintve: 2021.11.09.)
65. Dr. Richlach Mónika: *Az oktatási célú szabad felhasználás.* Merosus, 2020.04.14.,  
<https://merosus.hu/foto-jog/2020/04/az-oktatasi-celu-szabad-felhasznalas/> (utoljára megtekintve: 2021.11.09.)
66. Paál Vince: *Az EU Bírósága az internetes fotók jogszerű felhasználásáról döntött.* Nemzeti Média- és Hírközlési Hatóság Médiatanács Médiatudományi Intézete,  
[https://nmhh.hu/cikk/197636/Az\\_EU\\_Birosaga\\_az\\_internetes\\_fotok\\_jogszeru\\_felhasznalasrol\\_dontott](https://nmhh.hu/cikk/197636/Az_EU_Birosaga_az_internetes_fotok_jogszeru_felhasznalasrol_dontott) (utoljára megtekintve: 2021.11.09.)
67. Ronai András: *Jerusalema kibívás: miért kell engedélyt kérni, mikor kell fizetni, kit kell keresni?* 2021.05.07.,  
<https://dalszerzo.hu/2021/05/07/jerusalema-oromtanc-kihivas/> (utoljára megtekintve: 2021.11.09.)
68. *Honnan szerezhetünk jogtisztán zenéket, hangokat, zörejekeket?*  
[https://buvosvolgy.hu/tudastar/cikk/Honnan\\_szereshetunk\\_jogtisztan\\_zeneket\\_hangokat\\_zorejeket?fbclid=IwAR1\\_K1WoCtI6e8\\_fEsymp-9ZcWXGhVBMz3sX0EecOrIuGXyHDmVYiFWsnR8](https://buvosvolgy.hu/tudastar/cikk/Honnan_szereshetunk_jogtisztan_zeneket_hangokat_zorejeket?fbclid=IwAR1_K1WoCtI6e8_fEsymp-9ZcWXGhVBMz3sX0EecOrIuGXyHDmVYiFWsnR8) (utoljára megtekintve: 2021.11.09.)
69. *Facebook Hanggyűjtemény.*  
[https://business.facebook.com/creatorstudio/?tab=ct\\_sound\\_collection&collection\\_id=all\\_pages](https://business.facebook.com/creatorstudio/?tab=ct_sound_collection&collection_id=all_pages) (utoljára megtekintve: 2021.11.09.)
70. *A szerzői joggal kapcsolatban gyakran ismételt kérdések.*  
<https://euipo.europa.eu/ohimportal/hu/web/observatory/faqs-on-copyright-hu> (utoljára megtekintve: 2021.11.09.)





# Digitális eszközökkel támogatott verstani kutatások

Gortva Tamás<sup>1</sup>, Kelemen Péter<sup>2</sup>, Kelemen András<sup>3</sup>

<sup>1</sup>tgortva@gmail.hu

SZTE BTK

<sup>2</sup>peter.balazs.kelemen@gmail.com

ELTE IK

<sup>3</sup>kelemen.andras.felix@szte.hu

SZTE JGYPK

**Absztrakt.** Az 1960-as évektől ismét a nyelv- és irodalomtudomány homlokterébe kerülő műfajkutatások hajtómotorja a kommunikáció matematikai leírásának megvalósulása [1], és az ezzel szinte egy időben zajló *pragmatikai fordulat* [2] voltak. Továbbá a digitális bölcsészet egyik alapvető elvárása a *forrásanyag feldolgozásához szükséges folyamatok automatizálhatósága, ami minőségében és látásmódját tekintve újszerűen kutatható adatbázist eredményez* [3]. Jelen projekt e kutatásokhoz kíván hozzászólni a verstani elemzés és a stilometria számítógéppel automatizálható eszköztárának fejlesztése révén. Ezek a szoftverek végső soron az egyes szövegek objektív összehasonlíthatóságának egyik alapfeltételét teremtik meg, hosszú távon pedig hozzájárulhatnak a korpuszban lévő szövegek hálózatainak feltáráshoz, tehát a stilisztikai-retorikai (részben műfaji) sajátosságok leírásához.

**Kulcsszavak:** stilometria, verstan, digitális bölcsészet, szövegfeldolgozás, algoritmus

## 1. Bevezetés

Az 1960-as évektől ismét a nyelv- és irodalomtudomány homlokterébe kerülő műfajkutatások hajtómotorja a kommunikáció matematikai leírásának megvalósulása (Shannon – Weaver 1964), és az ezzel szinte egy időben zajló *pragmatikai fordulat* (Tátrai 2011: 279) voltak. Ugyancsak erre az időre tehető a *bálozat* kutatás megújulása is, melynek ma egyik – sokat vitatott, a bölcsészettudomány terén alkalmazhatónak látszó – elméleti kerete a skálafüggetlen hálózatok ábrázolása (Barabási 2002, Csermely 2005), mely az informatikai technológia bevonása nélkül elképzelhetetlen. Továbbá a digitális bölcsészet egyik alapvető elvárása a *forrásanyag feldolgozásához szükséges folyamatok automatizálhatósága, ami minőségében és látásmódját tekintve újszerűen kutatható adatbázist eredményez* (Sennyei 2018: 115–116).

Jelen projekt e kutatásokhoz kíván hozzászólni a verstani elemzés és a stilometria számítógéppel automatizálható eszköztárának fejlesztése révén. A lírai nyelvhasználat felszíni struktúrája (parole; nyelvhasználata) és mélyszerkezete (langue; pragmatikai és szemantikai meghatározottsága) között mintegy átmenetet képez a verstan, amely egyszerre kíván megfelelni a nyelv diktálta szabályszerűségeknél, és a nyelvhasználat elvárásainak, kommunikációs szándékának. E jellemzők adatolható feltárása végső soron az egyes szövegek objektív összehasonlíthatóságának egyik alapfeltételét teremtik meg, hosszú távon pedig hozzájárulhatnak a korpuszban lévő szövegek hálózatainak feltáráshoz, tehát a stilisztikai-retorikai (részben műfaji) sajátosságok leírásához. Kutatásunk szövegtörzsét a gimnáziumi tananyag versanyaga szolgáltatja, hiszen e szövegek magas kanonizáltságuk és közis-

mertségük által nagyban hozzájárulnak szövegfogalmunk és stilisztikai-retorikai érzékünk kialakulásához.

## 2. Időmértékes verselés

Az időmértékes verselés nem más, mint a hosszú és rövid szótagok – jobban mondva morák – szabályos ismétlődésén alapuló versrendszer. Egy szótagot rövidnek nevezünk akkor, ha benne a magánhangzó rövid és utána maximum egy mássalhangzó áll. Verstani jelölése: **U**. Egy szótag hosszú, ha a benne található magánhangzó hosszú, és/vagy rövid, bár utána több mint egy mássalhangzó áll. Verstani jele: **—**. (A sorvégi szótagok – annak függvényében, hogy a következő sor elején mássalhangzó vagy magánhangzó áll – lehetnek rövidek vagy hosszúak egyaránt.)

### 2.1. Felépítése

Az időmértékes verselés alapegysége a **mora**. Egy morának egy rövid szótag felel meg. A hosszú szótag két mora. A verslábak morákból épülnek fel. A **versláb**akat annak alapján különböztetjük meg, hogy hány morásak. A leggyakoribb verslábakat az 1. táblázat tartalmazza.

Név	Versláb
Jambus	U —
Trocheus	— U
Spondeus	— —
Daktilus	— U U
Anapesztus	U U —

1. táblázat: Verslábak

A verslábak szabályos ismétlődése (Szepes – Szerdahelyi (1981) nyomán) olyan állandósult metrikai képletté szilárdul, amely – az állandósult szókapcsolatokhoz hasonló módon – a legkülönfélébb sorokba, periódusokba épül be, s így mintegy a versek „metrikai közhelyévé” válik, tehát a versláb nál nagyobb, a periódusnál és a versornál kisebb metrikai egységgé válik. Ezt a kettő vagy több versláb alkotta állandósult kapcsolatot nevezzük **kolón**nak, amely önállóan is létrehozhat ugyan verssort (legalábbis sortöredéket), azonban két vagy több kolón, illetve sor állandósult kapcsolata olyan igen gyakori, széles körben használt metrikai képletet szilárdíthat meg egy adott versrendszerben, amely rendre ugyanabban a formában jelentkező **periódusként** határozható meg, és amely akár verssorokban, akár azok egy szakaszában vagy azokon túlnyúlóan is megjelenhet. Felületesen szemlélve a **verssor** az az egység, amelyet a költő egy sorba ír, azonban a szövegek mélyebb struktúráját szemlélve könnyen belátható, hogy ez szükséges, de nem elégséges definíció. Minthogy a verssorok több esetben a prózai sorokhoz hasonlóan egymásba hajlanak (enjambement) vagy – mint a XVIII. század előtt sok esetben – folyószöveggént lejegyzett ritmikai egységek, így a verssor valójában nem ritmikai egység, a látási forma kialakítása révén csupán hozzájárul a ritmusérzet kialakulásához. Ugyancsak az alkotó egyéni leleménye és a versritmus által diktált elvárás közé helyezhető a **vers-**

**szak** vagy **strófa** egysége, amely a vers olyan két vagy több sorból álló, önmagában zárt metrikai egysége, amely a szöveg nagyobb részében vagy egészében szabályszerűen ismétlődik.

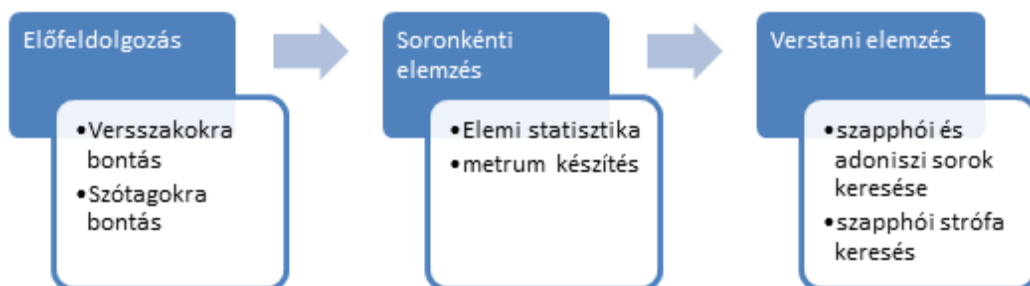
## 2.2. Sorok, strófák, versformák

A *daktilikus* hexameter 23-24 moraértékű, hat verslábból álló, antik eredetű sorfajta, melynek első négy lába lehet daktilus vagy spondeus, az ötödik kötelezően daktilus, a hatodik trocheus vagy (a következő sor kezdőbetűjétől is függően) spondeus. Ez a sor valójában felfogható két kolón állandósult kapcsolataként, melynek legjellemzőbb periódusa az utolsó két lábat alkotó daktilus és a hosszú vagy közömbös szótagot tartalmazó spondeus vagy trocheus: — U U / — U . Ez a kolón visszaköszön számos sorban, és önmagában **adoniszi sorként** ismeretes, többek között a szapphói strófa zárószaként is azonosítható.

A szapphói strófa (hagyományos felfogás szerint) három tizenegy szótagos, jellemzően öt verslábból fölépülő, ún. **szapphói sorból** és egy adoniszi sorból áll. További megkötés a szapphói sor kapcsán, hogy az öt verslábból a középső (harmadik) mindig daktilus, a többi pedig jellemzően trocheus, mely(ek) spondeussal felcserélhetők.

## 3. Algoritmus

A következőkben az időmértékes verselemzés egy általunk létrehozott számítógépes algoritmusát és annak implementációját mutatjuk be. Az elemző bemenetként folyó szöveget vár. Ez lehet maga a teljes vers vagy annak részlete. Amennyiben a versszakokat az elemzőben is szeparáltan szeretnénk látni, akkor a bemenetben a versszakokat üres sorral kell egymástól elválasztani. Az elemző jelenlegi verziójában a következő műveleteket végzi (1. ábra):



1. ábra: Az elemző algoritmus folyamatábrája.

Részletezve:

1. az egyes verssorokat a magánhangzók mentén szótagokra bontja. A bontással egy időben kategorizálja a szótagot (hosszú, rövid, magas, mély hangrendű).
2. Minden egyes verssornál megszámolja rövid és a hosszú szótagok számát.
3. Meghatározza az adott verssor milyen verslábakkból áll (metrum).
4. A metrum ismeretében eldönti, hogy a verssor a szapphói vagy adoniszi, vagy valami más.
5. Amennyiben a versszak 4 sorból áll, akkor a sorok metrumának ismeretében eldönti, hogy a versszak szapphói strófa-e.

### 3.1. Implementáció

Az elemző az internetes használhatóság miatt Java szervletként lett implementálva. A 2. ábrán a rezponzív felhasználói felület látható. A „Versválasztás” menüpont szolgál a versek feltöltésére a lokális gépről a szövegdobozba. Az „Elemi statisztika” menüponttal lehet az elemzést elindítani.

The screenshot shows a web application interface. On the left is a red sidebar with the title 'Verselemző' and three menu items: 'Versválasztás', 'Elemi statisztika', and 'Elemi statisztika'. The main content area is white and contains two sections. The first section is titled 'Vers' and shows a text input field with a red border containing a poem snippet with red underlines under each word. The second section is titled 'Statisztika' and displays analysis results for the first stanza, including the number of long and short words, and a syllable count visualization for the first line of the poem.

## Vers

```
Partra szállottam, levonom vitorlám,  
A szelek mérgét nemesen kiálltam.  
Sok Charybdís közt, sokezer veszélyben  
Izzada orcám.
```

## Statisztika

1. versszak

Hosszú szótagok: 7 Rövid szótagok: 4

P'artr'a sz'áll'ott'am, l'ev'on'om v'it'orl'ám,

\_ U \_ \_ \_ U U \_ U \_ \_ szapphai sor

**2. ábra:** Az elemző felhasználói felülete.

#### 3.1.1. Adatábrázolás

A szoftver a verset versszakonként egy listában tárolja. A versszakokon belül listában tároljuk a verssorokat. Mivel soronkénti elemzést végzünk, minden egyes verssorhoz tároljuk a sor szótag szerinti felbontását egy szótag listában, valamint egy sztringben a metrumát.

### 3.1.2. Publikus metódusok

Az alábbiakban bemutatjuk az egyes osztályok publikus metódusait

VERS	
Metódus	Funkció
<code>void felbont()</code>	Felbontja a verset az üres sorok mentén versszakokra
<code>void elemez(PrintWriter out)</code>	Elvégzi az elemzést és az eredményt megjeleníti a weboldalon

VERSSZAK	
Metódus	Funkció
<code>void verszakToSzotag()</code>	Soronként szótagokra bontja a versszakot.
<code>void sorToSzotag(String txt)</code>	Magánhangzók mentén szótagokra bontja a verssort.
<code>String sorMetrum(String txt)</code>	Meghatározza egy verssor metrumát
<code>void Metrum()</code>	Soronként meghatározza a versszak metrumát.
<code>boolean isAdonisziSor(String sor)</code>	Eldönti egy sorról, hogy adoniszi-e
<code>boolean isSzapphoiSor(String sor)</code>	Eldönti, egy sorról, hogy szapphói-e
<code>boolean isSzappoiStrofa()</code>	Eldönti, hogy a verssor szapphói strófa-e
<code>int hosszuSzotag(String sor)</code>	Megadja, hogy hány hosszú szótag van a verssorban
<code>int rovidSzotag(String sor)</code>	Megadja, hogy hány rövid szótag van a verssorban

## Következtetések

A természetes szövegek számítógépes felismerése a neurális hálókön alapuló gépi tanulás alkalmazásával az elmúlt pár évben rendkívül látványos fejlődésen ment keresztül. Ez a fejlődés megteremtette az irodalomtudományban és a nyelvészetben történő alkalmazás lehetőségét is. Egy szövegfelismerő rendszer gépi tanítása lényegében pozitív és negatív példákon hosszú iterációkkal történik. Ehhez nagyméretű pozitív és negatív korpuszra és vele együtt jelentős számítógépes kapacitásra van szükség. Meglátásunk szerint a verstanban ezt a hosszú és drága tanítási folyamatot algoritmikus eszközökkel drasztikusan le lehet rövidíteni. Ez egyben a felhasználó részéről sokkal kisebb számítástechnikai erőforrást igényel.

## Irodalom

1. SHANNON, Claude E. – WEAVER, Warren (1964): *Theory of Communication*. The University of Illinois Press, Urbana
2. TÁTRAI Szilárd (2011): *Pragmatika*. In: BALÁZS Géza (szerk.) (2011) *Nyelvészetről mindenkinek, 77 nyelvészeti összefoglaló*. Inter, Bp.
3. SENNYEI, Pongrácz (2018): *Viták és viziók a digitális bölcsészetről*. In: *Digitális Bölcsészet 1* (szeptember), 111-120.  
<https://doi.org/10.31400/dh-hun.2018.1.228>.
4. SZEPEs Erika – SZERDAHELYI István (1981): *Verstan*. Gondolat, Bp.

# Offline szituációs játék informatikai rendszerek oktatásához

Korom Szilárd<sup>1</sup>, Illés Zoltán<sup>2</sup>

<sup>1</sup>korom.szilard@inf.elte.hu, <sup>2</sup>illes@inf.elte.hu

ELTE IK

**Absztrakt.** Már számos informatikai fogalom, tananyagelem oktatására született olyan offline, a valóságban játszható módszertan, mely segíti a diákokat a megértésben. Például rendezési algoritmusokra, számrendszerekre vagy adatstruktúrákra. Az alábbiakban egy olyan szituációs játék kereteit kívánjuk bemutatni, mely a szoftverrendszerekre, hálózati kommunikációra fókuszál úgy, hogy különböző programozható eszközök együttes működését kell szimulálni. Reményeink szerint a játék izgalmas módon mozgatja meg a diákokat, miközben nehéz, de annál fontosabb informatikai fogalmak megértését segít az általános és középiskolás korosztályban. A játék egy okos otthon szimulációja tantermi környezetben, mindenféle informatikai eszköz használata nélkül.

**Kulcsszavak:** CSUnplugged, számítógép nélkül, okosotthon, informatikai rendszerek, hálózati kommunikáció, offline játék

## 1. Bevezetés

A számítógép nélküli oktatás lényege, hogy a diákok játékosan szerezhessenek valós tapasztalatot a lényeges informatikai fogalmakról. A módszerrel szembeni legfontosabb követelmények, hogy ne kelljen hozzájuk semmilyen digitális eszköz, illetve, hogy minimálisak legyenek az előkövetelmények. A lényeg tehát, hogy olyan problémákat kelljen eljátszaniuk, melyek megoldásához nagyrésztben a saját intuícióikra hagyatkozhatnak.

Ezen játékok „központján” [1] rengeteg példát találhatunk, összesen több mint 20 nyelvre lefordítva. A „CSUnplugged” kifejezés az 1990-es évekre megy vissza. Az első komoly kötet Tim Bell, Mike Fellows és Ian Witten szerzők által írt “Computer Science Unplugged: Off-line activities and games for all ages” kötet volt [2]. A játékok közül sok magyar nyelven is elérhető [3]. A témában számos cikk is készült az évek folyamán [4, 5, 6], melyek nagyon változatos témákat fednek le az informatikán belül. A rendezési algoritmusok például egyáltalán nem könnyűek, azonban létezik több igen egyszerű, tantermi körülmények között is játszható gyakorlata [7]. Például, ha a diákokat megkérjük, valószínűleg igen hamar sorba tudnak állni magasság szerint. A tényleges algoritmus valószínűleg nem tudatosul bennük, de ha megfigyelik a saját tevékenységüket, a használt módszer formalizálható. A rendezési algoritmusok többsége pedig a valóságban is eljátszható, így a tanár felvethet, körbeírhat módszereket, instrukciókkal telergetheti a diákokat, hogy máshogyan is kipróbálják, hogyan lehetne sorba állni.

Az offline játékoknak mára már kiterjedt szakirodalma van. Természetesen tantárgyakon átívelve is alkalmazható [8]. Természetesen nem kell/lehet mindent lecserélni ilyenre, de mindenképpen a tananyagba illeszhető. Ez egyébként egyes tankönyvek esetében már meg is történt [9, 10], illetve kódolással foglalkozó tanulást támogató rendszerek is átvették, mint például a code.org [11]. Persze az, hogy sokan használják, nem jelenti azt, hogy előnyös is. Mára már azonban kijelenthetjük, hogy a módszer bizonyítottan hatékony [12].

## 2. A szituációs játék ismertetése

A szituáció az, hogy egy tanteremben vagyunk a diákokkal és elképzeljük, hogy kiépítünk egy okosotthon projektet, ahol az informatikai eszközök, a vezérlők, szenzorok, átviteli eszközök, átjárók és megjelenítési megoldások (monitorozás) mind-mind diákok. Például a terem különböző pontjain szeretnénk mérni a hőmérsékletet, akkor oda kell állnia a diáknak, aki „méri” a hőmérsékletet (valóságban kitalál egy számot, ami hozzávetőlegesen reális), egy másik diák a kapcsolónál áll és a lámpát irányítja, megint másik diák a táblára tudja felírni az értékeket. A játék célja, hogy minél hatékonyabb felosztást és kommunikációs formákat találjanak ki a rendszer működtetésére. A kulcs az, hogy a rendszer bővítésével, bonyolításával milyen módon kell a különböző „komponenseket módosítani”. Miközben a feladat mélyül és a diákok új és új megoldásokat találnak ki, új szereplőket vonnak be a játék egy szoftver architektúrájává válik át. A diákok fejében persze nem (feltétlenül) fogalmazódik meg a konkrét informatikai analógia. Erre a folyamatos reflexió és a tanári terelgetés enged teret. Például a diákok által eljátszott szerepek elemzésével megválaszolható, hogy ezt hány program, hány hardver tudná megvalósítani.

## 3. Informatikai háttér

Először be kívánjuk mutatni, hogy konkrétan milyen informatikai eszközök és ismeretek szükségese ahhoz, hogy ténylegesen leimplementáljunk egy megoldást a fent említett problémára. A bemutatás nem teljessé, s nem részletekbe menő, hiszen az kívül esik a cikk keretein, de egy átfogó képet kívánunk adni.

### 3.1 Szenzorok/vezérlők

A szenzorok és vezérlők olyan alacsony-szintű informatikai eszközök melyek képesek jelet fogni, vagy küldeni a valós világban. Például hőmérséklet, fény vagy távolság érzékelő szenzor, de ide értünk egy egyszerű LED-et, vagy kapcsolót, esetleg motort is. Ezen a szinten nem programozható célszámítógépekről van szó, ezeknek az eszközöknek szükségük van egy közvetlen vezérlőre.

### 3.2 Helyi feldolgozás

Az előző réteg vezérlőiről van szó, melyek elsősorban M2M kommunikációra alkalmasak, vagyis vagy folyamatos futás mellett biztosítanak valamiféle értéket, vagy egy irányított kérésre cselekednek (pl.: kapcsolnak fel egy lámpát, küldenek vissza adatok).

Ezen a szinten beszélhetünk áramköri megoldásokról is, hiszen a perifériákkal valahogyan össze kell kötni a vezérlőt. Természetesen „ready-to-use” eszközök is a rendelkezésünkre állnak GPIO-val ellátva (Általános célú be- és kimeneti kapcsolatok). A legelterjedtebb ezek közül, melyek oktatási célokra is felhasználható az Arduino és a Raspberry Pi számítógépek. Előbbivel beágyazott-rendszereket, utóbbival magasabb-szintű, asztali számítógép működéséhez nagyon hasonló termékeket lehet készíteni.

### 3.3 Hálózat és internet, adattovábbítás

A kommunikációs réteg legfontosabb kérdése, hogy mit jelent a kommunikáció. Az IoT eszközök esetében a hálózati protokoll adja meg a választ. A leggyakoribbak talán:

- MQTT vagy AMQP
- HTTP vagy REST

#### 3.3.1 Az MQTT és az AMQP protokoll

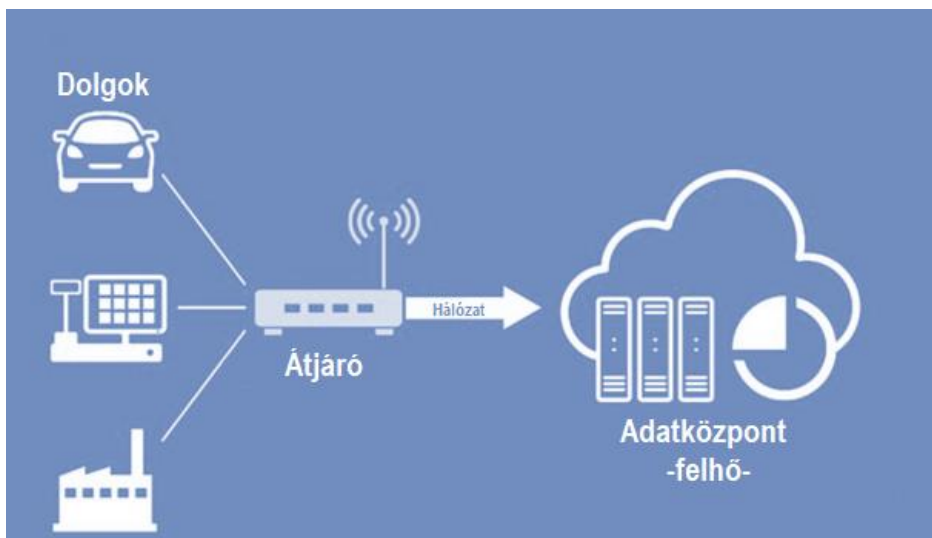
Ezeket a protokollokat (illetve ezt a típust) azért is tartjuk fontosnak kihangsúlyozni, mert a valós életbeli kommunikáció talán erre, a publish-subscribe mintára hasonlít a leginkább. Ez a protokoll



feliratkozásokról és publikálásokról szól. Az ötlet az, hogy aki az üzenetet továbbítja nem foglalkozik azzal, hogy ki fogja fogadni az üzenetet, mondhatni csak elkiáltja magát, s aki feliratkozott (aki figyel) majd megkapja, s feldolgozza az üzenetet. A működési elv az, hogy van egy központi szoftver (úgynevezett broker), aki fogadja a publikált adatokat és elküldi azoknak, akik az adott fajta üzenetre feliratkoztak. Ez tehát merőben eltér a kérés-válasz struktúrától, hiszen aki az adatot küldi, az ezt nem kérésre teszi, ráadásul a kliens nem is tudja, hogy azt ki fogja fogadni (és hogy aki akarta, az ténylegesen megkapta-e). Általánosságban ez a sokeszközös világban igen elterjedt, így az okosothon megvalósításoknál is. Például egy hőmérséklet érzékelő szenzornak nem feltétlenül kell tudnia róla, hogy ki kíváncsi az adatra. A szenzor továbbítja egy központi alkalmazásnak (az okosothon vezérlőnek), amiktől adatokat kérnek a felhasználók egy webes felületen, vagy egy mobil alkalmazás segítségével.

### 3.4 Általános IoT architektúra

Egy valós IoT megoldás általános architektúrája tehát az 1. ábrán látható:

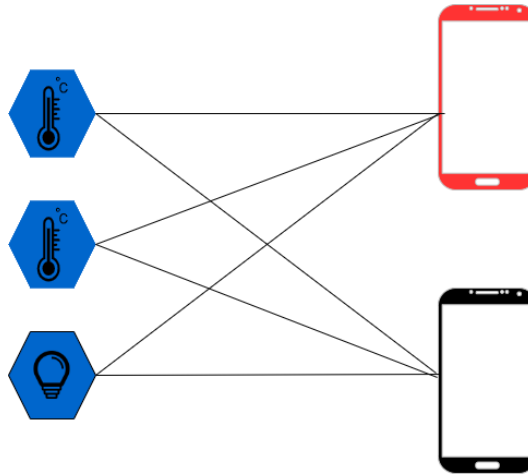


1. ábra: IoT megoldások általános architektúrája

### 3.5 Konkrét architektúra

A továbbiakban azt szeretnénk bemutatni, hogy a mi konkrét okosothon megoldásunknak az architektúrája hogyan nézhet ki. Nem kívánjuk teljesen részletezni őket, hiszen nem ez a cikk fő fókusza. Természetesen más megoldások is elképzelhetők. A példák azért fontosak, mert tulajdonképpen ezeket szeretnénk eljátszatani a diákokkal. A játék során pedig a diákoktól azt várjuk el, hogy ezeket tervezzék meg a saját tapasztalatok alapján.

### 3.5.1 Okosotthon architektúra szerver nélkül



2. ábra: Okosotthon architektúra szerver nélküli változata

Ebben az esetben összesen 3 alkalmazást kell készítenünk:

- A hőmérséklet érzékelők beágyazott rendszerek, hiszen csak adatot mérnek és küldenek tovább. Alacsony szintű program fut rajtuk.
- A megvilágítás szintén beágyazott rendszer, hiszen csak egy egyszerű kérést kell kiszolgálnia és kezelnie. Alacsony szintű program fut rajtuk.
- Mobil alkalmazás, mely képes feltérképezni a hálózaton elérhető eszközöket, képes velük kommunikálni. Mivel a mobil eszközön van operációs rendszer, nem alacsony szintű programról van szó. Megjelenéssel itt foglalkozni kell.

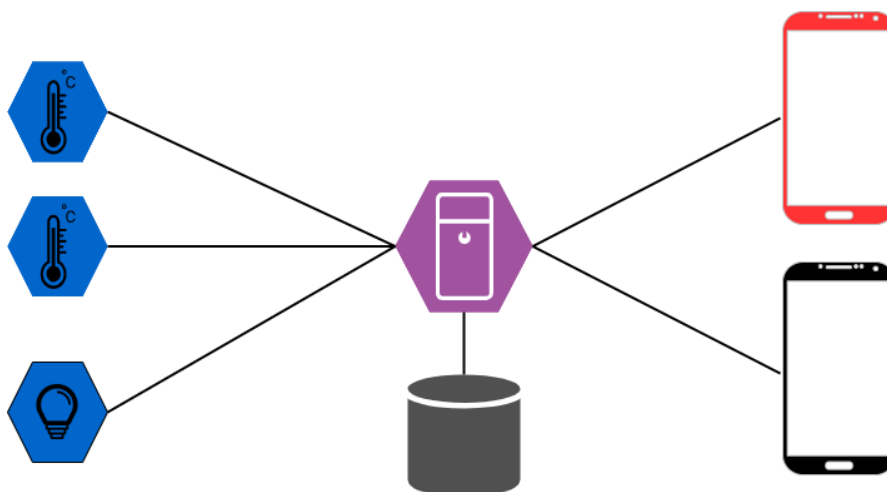
Ezzel a megoldással találkozhatunk a leggyakrabban a normál felhasználói környezetben. Előnye, hogy a megvalósítás olcsó és rugalmas, hiszen nem igényli egy szerver működtetését és fenntartását, integrálása könnyű. Számos olyan eszközzel találkozhatunk a piacon, amely megvétel után egy alkalmazás letöltését követően már működőképes és használható.

Hátránya, hogy a rendszer nem integrált. A valóságban ez azt jelenti, hogy ahányféle eszközt veszünk, annyiféle alkalmazást kell letöltenünk. Az eszközökkel való kommunikáció erősen korlátozott, ha sok kliens szeretnénk működtetni, más problémákba is ütközhetünk. Ha komplexebb megoldást keresünk, ahol már az eszközök együttes működése szükséges döntési logikával, akkor az architektúra alkalmazhatatlan (például azt szeretnénk, hogy a hőmérséklet függvényében kapcsoljon be a fűtés).

Mivel az eszközök teljesen különállóak, annyiféle alkalmazást kell fejleszteni ahányféle kliensünk van. Ha például szeretnénk webes felületen számítógéppel elérni az eszközöket, fejleszteniünk kell egy olyan alkalmazást, mely minden lehetséges komponenssel tud kommunikálni. Ha egy új fajta szenzort építünk a házba, azt megint külön kezelni kell.

További hátrány, hogy a szerver hiányában minden egyes eszköznek magának kell gondoskodnia a biztonságról. Az IoT világ egyik legnagyobb problémája a biztonság megfelelő fenntartása [13].

### 3.5.3 Okosotthon architektúra szerverrel



3. ábra: Okosotthon architektúra szerveres változata

Ebben az esetben 4 alkalmazást kell készítenünk. A korábbi hármat, valamint egy szervert. A szerverprogram képes a szenzorokkal és vezérlőkkel kommunikálni, döntéseket hozni, valamint adatot kiszolgálni és tárolni (például egy adatbázisban), megjelenés nem szükséges. A szerverprogram semmiképpen nem alacsony szintű, hiszen több szálon is kommunikálnia kell és változatos feladatokat el kell látnia.

Előnye, hogy megfelelő tervezés esetében nagymértékben függetleníthető attól, hogy kitől kap és ad adatot. A rendszer tehát skálázható és könnyen bővíthető újfajta eszközökkel.

A rendszer továbbá biztonságos, hiszen a szenzoroknak csak egyetlen eszközzel kell kommunikálniuk, mondhatni, hogy elfedi az alsó réteget a struktúra.

Hátránya, hogy mind a készítése, mind a fenntartása költséges, hiszen az eszközök egyáltalán nem függetlenek egymástól. Úgy kell megválasztani minden egyes komponenst, hogy azok alkalmazsak legyenek egy saját szervermegoldással való integráláshoz. Szükség van továbbá egy szervergépre, melyen a programunk fut, melynek állandóan elérhetőnek kell lennie.

## 4. A játék leírása

Az informatikai rendszerek működésének eljátszási problémája, hogy egy program, egy valós eszköz merőben eltér attól, hogy mi emberek a valóságban hogyan cselekednénk. Ha például kijelölünk valakit, aki a terem egy általunk választott pontján méri a hőmérsékletet, majd az eredményt fel kell írnia a táblára, akkor valószínűleg ezt úgy oldaná meg, hogy kitalál egy számot, majd odasétál a táblához és felírja. Informatikai rendszerek analógiájában azonban annál sokkal egyszerűbb megoldások is léteznek, minthogy építsünk egy programot és robotot, mely képes odasétálni és még rajzolni is. A probléma felvetése azonban nem hasztalan, hiszen ez rámutat az informatikai és hétköznapi gondolkodás különbségére.

A fő probléma tehát az, hogy a kommunikáció és az érzékelés merőben más gépek esetében. Például a diáknak nem tudunk olyan instrukciót adni, hogy ne lássa, hallja a többieket, de mégis el tudjon olvasni bizonyos üzeneteket.

A végső cél tehát hogy a diákokkal, mint szereplőkkel egy olyan rendszert építsünk fel, mely egy valós okosotthon informatikai megoldását modellezi.

#### 4.1 Módszertan

A tervezést a diákoknak kell csinálniuk, s bonyolultsága fokozatosan mélyül, rétegről rétegre. A módszer alapja a probléma felvetése, majd kérdésekkel való terelgetése. Egy lehetséges kérdéssorozatot mutat be az 1. táblázat.

Felvetés - javítás	Várt kimenet	Játék	Informatikai kapcsolat
Milyen szereplőkre van szükségünk, ha szeretnénk a terem 2 sarkában mérni a hőmérsékletet, szeretnénk a lámpát kapcsolgatni, szeretnénk, hogy egy felhasználó a táblánál tudja ezt vezérelni, mintha a táblán lévő rajz lenne maga a vezérlő alkalmazás?	Valaki megtervezi hogyan nézzen ki az alkalmazás (felrajzolja a táblára). A hőmérsékletmérők 1-1 diákok, valaki a kapcsolónál áll. Létezik felhasználó, aki megnyomja a táblán lévő rajzolt gombot, vagy elkiáltja magát, ha adatra van szüksége.	Mindenki nézi, hogy a táblánál lévő felhasználó mit szeretne (megnyom egy gombot, kér valamit), s külön-külön reagálnak rá.	Kérés-válasz protokollok (pl.: http). Több eszköz létezik különböző céllal. Grafikus megjelenést is tervezni kell. Létezik hálózat. Felvethető a peer-to-peer kapcsolat.
A probléma az előző rendszerrel, hogy valójában a gombot nem láthatják az érzéklők/vezérlők és valójában beszélni sem tudnak.	Kijelölnek egy új diákot, aki az alkalmazást vezérli és továbbítja a kérést a vezérlők felé, illetve felrajzolja, ha érkezik információ.	Ha egy felhasználó megnyomja a gombot, már a vezérlő diák beszél és csak a kijelölt „céleszközzel” és ugyanez fordítva.	Szerver-kliens kapcsolat, de még adatbázis nélkül. Publish-subscribe alapú kommunikáció.
Folyamatosan szeretnénk látni a hőmérsékletet.	Az érzékelő diákok folyamatosan hangosan mondják mi van náluk.	Az eredmény túl káosz, nehezen kivehető, ráadásul a rajzoló diák nem tudja követni.	Adatovábbító réteg, gateway, queue. A folyamatos adatmondogatás felel meg az MQTT/AMQP protokolloknak.
Oldjuk meg, hogy ne legyen hangzavar, ne akadjanak össze az információk.	Kijelölnek adattovábbító diákokat, akik viszik az információt az érzéklők és a monitorozó diák között.	Az adattovábbítók viszik az információt, így nincs hangzavar. Ha egyszerre érnek oda, sorba kell állniuk.	Nagyon szigorú architektúráris és kommunikációs szabályok, a zaj kiszűrésére és a hatékony adattovábbításra. Minden egyes komponensnek nagyon jól behatárolt feladata van.
Több vezérlő és felhasználó legyen (akár egy tényleg okosotthon projektnél).	A diákok kettéosztják a táblát és bevonnak egy új vezérlő embert, adattovábbító embereket.	A kommunikáció lelassul, mert a szenzoroknak és kapcsolóknak több adatot kell szolgáltatni.	

1. táblázat: Egy lehetséges menete a játéknak

Konkrét útmutatót a játékkal kapcsolatban nehéz adni, mert az egyes felvetések attól függenek, hogy a csoport a korábbira milyen megoldást ad. Elképzelhető például, hogy a diákok már a legelején egy elég jóltervezett struktúrát adnak:

- Van vezérlő ember
- Vannak adattovábbítók
- Van rajzoló ember
- A kommunikáció fegyelmezetten és a szabályok szerint történik

Persze ebben az esetben is lehet még fejleszteni, új komponenseket illeszteni a rendszerbe, de a folyamat menetét akár meg is fordíthatjuk, hogy direkt egy rossz megoldás eljátszását kérjük tőlük az azt, hogy fogalmazzák meg miért nem működött ez a megoldás.

## 4.2 Hangsúlyok

A feladat eredeti formájában a hálózati kommunikációra és a szoftveres architektúrákra helyezi a hangsúlyt. Ez persze módosítható. A lényeg tehát az, hogy a tanár milyen kérdéseket tesz fel, merre terelgeti a diákokat:

- **Algoritmusok:** Pontosan milyen „logikát” kell az egyes diákoknak (szenzorok, kapcsolók, vezérlők, rajzoló) megvalósítaniuk?
- **Hardverek:** Pontosan milyen hardverekre lenne szükségünk, ha ezt ténylegesen meg szeretnénk csinálni? Mik a követelmények a hardverekkel szemben, mit kellene tudniuk?
- **Szoftverrendszer:** Milyen szoftvereket kellene írniuk, hogy ezt megvalósítsuk? Hogyan kezeljük szoftverszinten a felmerülő problémákat (adatvesztés, hibás adat, feltorló kérések)? Milyen jellegűek legyenek a megvalósítások, milyen paradigmákat kellene használni (pl.: konzolos, webes, mobilos, ablakos)? Szükség van-e operációs rendszerre az egyes hardvereken? Ha igen, milyenre? Ha nem, miért nem?
- **Adatok:** Milyen adatok tárolására van szükség? A tárolást és az adatok menedzselését ki végéze? Milyen adatsomagok menjenek át a kommunikáció során? Ezek az adatsomagok hogyan nézzenek ki?
- **Programnyelvek:** Milyen programozási nyelvek lennének a hasznosak a megvalósítás során? Mely programozási környezetekkel lehetne a különböző komponenseket megvalósítani? Milyen programozási paradigmákat használhatunk?

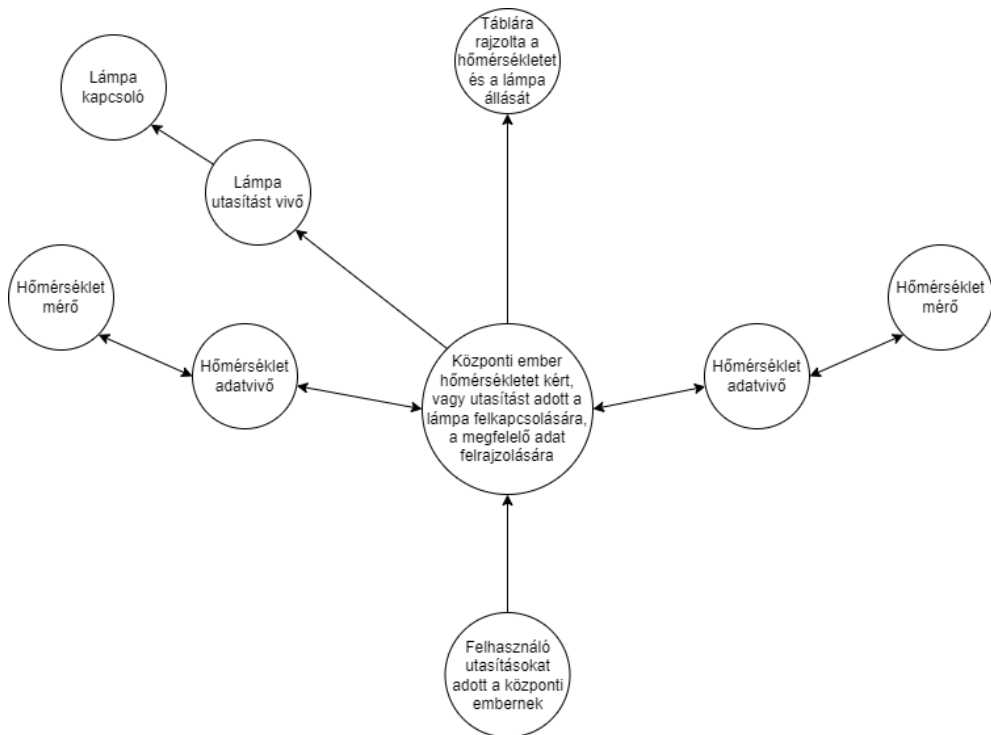
## 5. Éles tesztelés

### 5.1 A történetek leírása

A játékot kipróbáltuk az Eötvös Lóránd Tudományegyetem első féléves tanárszakos hallgatóival. A foglalkozáson 14 hallgató volt jelen és 60 percet vett igénybe. A foglalkozás végén egy kérdőívet is kitöltöttek.

A játék ebben a formában nyilván nem a legtermészetesebb, hiszen a tanárszakosok már informatika érettségivel rendelkeznek, tisztában vannak a játék által bemutatott fogalmakkal. Ettől függetlenül az ott nyert tapasztalat hasznos volt. A hallgatóktól egyszerre kértük a részvételt, mint „diákok”, s mint megfigyelő „tanárok”. A foglalkozás folyamán folyamatosan reflektáltunk a saját munkánkra és értékeltük mennyire volt hasznos az adott szekció.

A játék a szituáció bemutatásával, a feladat ismertetésével kezdődött. A közbeavatkozásunk nélkül a hallgatók nagyjából a következő architektúrát találták ki és valósították meg (minden kör egy embernek felel meg):



4. ábra: Hallgatók által kitalált első architektúra

A tervezéstől egészen odáig, hogy ténylegesen eljátszották és késznek tekintették a projektet nagyjából 20 perc telt el. Ezalatt közös ötletesés, illetve megbeszélés, finomítgatás történt, majd a tényleges játék. Nyilván az életkor és a szituáció miatt is nagyon fegyelmezettek és összeszedettek voltak, mely például egy általános vagy középiskolában nem várható el.

A foglalkozás közben nagyon kevés szerepünk volt. Amikor 1-1 fázissal készen voltak, adtunk egy következő fejlesztési lehetőséget, illetve tanulságot vontunk le velük közösen. (Hatékony volt-e így? Lehetne-e máshogy? stb.) Hogy konkrétan mi történt, a 2. táblázatban látható.

Felvetés - javítás	Tényleges kimenet	Tanulság	Informatikai kapcsolat
Lépjen be a játékba egy új felhasználó.	Az előzőkhez képest az az érdekes jelenség történt, hogy amikor a két felhasználó majdnem ugyanakkor kérte a lámpa felkapcsolását, akkor a központi embernek várnia kellett, mire visszaért a megfelelő adattovábbító ember.	A játék során ők problémának élték ezt meg, hiszen a valóságban itt hosszú másodpercek teltek el, de megbeszéljük, hogy persze lassabban vagyunk, mint ahogyan valóban mennek az adatok, de gépek esetében is történik ilyen feltorlódás, amit kezelni kell.	http (elsősorban a böngészéshez hasonlítottuk); szerver-kliens kapcsolat; 4 programot kellene írjunk, ha ténylegesen meg akarunk valósítani.
Legyen még egy megjelenítő (rajzoló).	Itt a központi ember már túl volt terhelve, nem tudta ellátni az összes feladatát, mert túl sok mindenkivel kellett volna egyszerre kommunikálnia.	A két rajzolást nem tudja egy ember csinálni még akkor sem, ha ugyanazt kell felrajzolni. A hallgatók körében felmerült, hogy ki kellene vezetni a központi embert a rendszerből.	Queue, megjelenítés teljes elkülönítése a szervertől.
Oldjuk meg adattovábbító ember nélkül (hangos beszéddel).	Sokan beszéltek egyszerre, de finomítgatók révén működött a rendszer.	A hallgatók érzékelték, hogy a probléma a kommunikáció. 1-2 valós teszt után nagyon szigorú szabályokat hoztak arra, hogy mikor ki beszélhet, illetve mindenkinek más hangszint adtak, hogy meg tudják különböztetni ki mondta. Utána levontuk a tanulságot, hogy valós szoftverek esetében is valahogyan erősen el kell különíteni az üzeneteket, kell küldeni egy azonosító adatot is, hogy tudjuk honnan érkezett a hőmérséklet.	Hálózati protokollok; adatsomagokra vonatkozó szabványok (adatleíró nyelvek); publish-subscribe protokoll.
Kérés nélkül folyamatosan lássuk a hőmérsékleteket a táblán.	A két hőmérséklet érzékelő ember hangosan mondogatta egyszerre az adatokat. Külön embert kellett behívni, aki csak a hőmérsékleteket írta folyamatosan.	Az előző szabályokat betartva elkülöníthetőek egymástól az adatok.	Többszálúság

2. táblázat: Éles teszt alatt történtek

Felvetés - javítás	Tényleges kimenet	Tanulság	Informatikai kapcsolat
Oldjuk meg kettő központi emberrel.	Ténylegesen eljátszani nem tudtuk, mert nem volt annyi ember, amennyire szükség lett volna.	Feleslegesen növeli a kommunikációk és a felelőségek számát.	-
Oldjuk meg központi ember nélkül.	Megvalósult, sok hallgató feladat nélkül maradt.	A publish-subscribe mód ebben az esetben hasznosabb, mint a kérdés-válasz szerinti. A rendszer rugalmasabb lett.	-

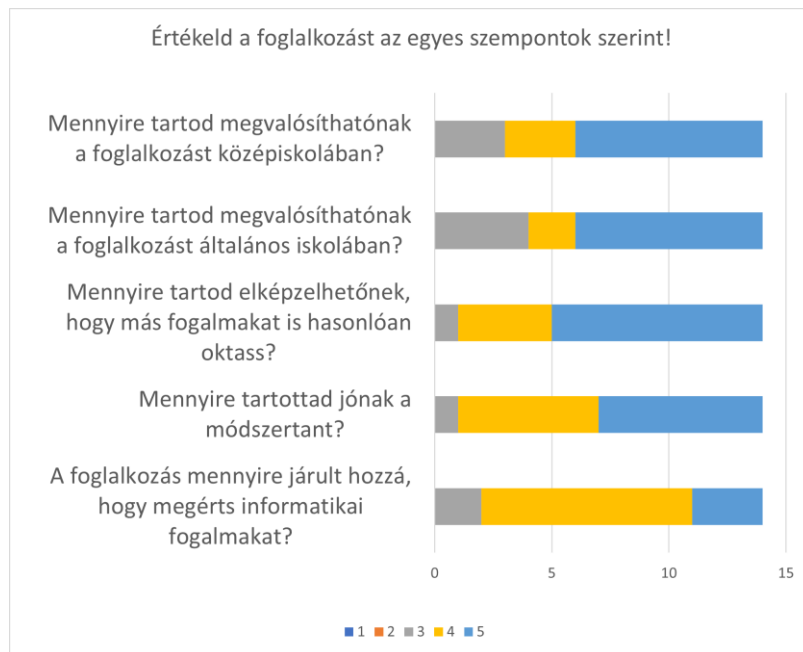
2. táblázat: Éles teszt alatt történtek

A foglalkozás összességében szerintünk jól alakult és a várt eredményt hozta. A felmerülő problémák elsősorban architektúráis vagy hálózati kommunikációs problémák voltak, s mindegyik analógiába állítható valamely informatikai fogalommal. A hallgatók (bár nem tudatosan), de szoftverarchitektúrát terveztek és igen kritikus, gyakori problémákkal szembesültek és adtak rá megoldást.

Nem utolsó sorban pedig a foglalkozás nagyon jókedvűen telt. A hallgatóknak együtt kellett dolgozniuk, közösen kellett megoldást találniuk és csak jól összehangolt munkával tudtak megoldást adni. Az egyes szituációk pedig gyakran viccesek voltak.

## 5.2 A kérdőívre adott válaszok

A következőkben a kérdőívben feltett kérdésekre adott válaszok láthatók.

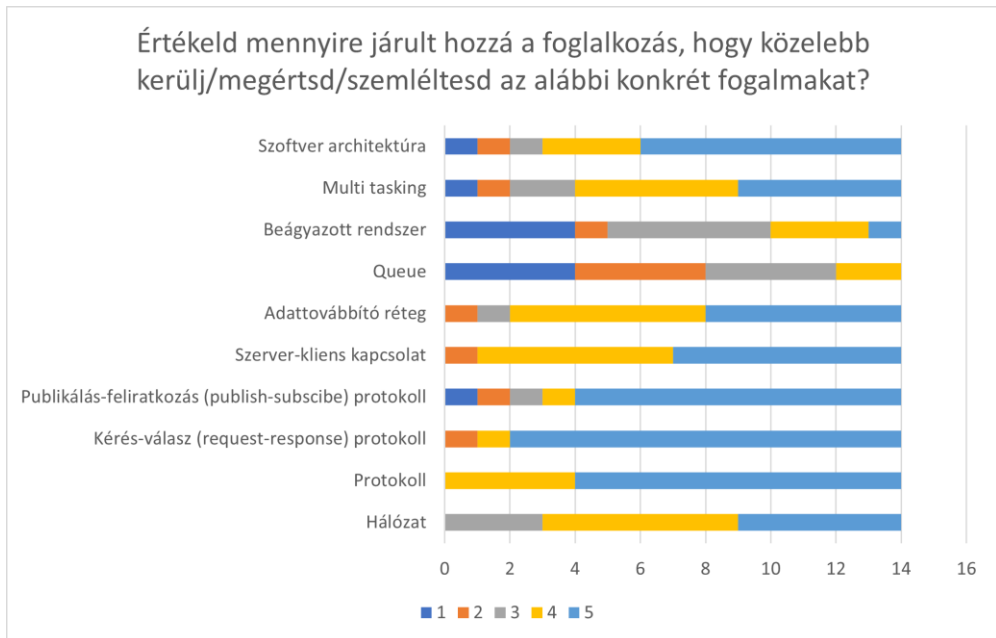


5. ábra: Értékelj a foglalkozást az egyes szempontok szerint!



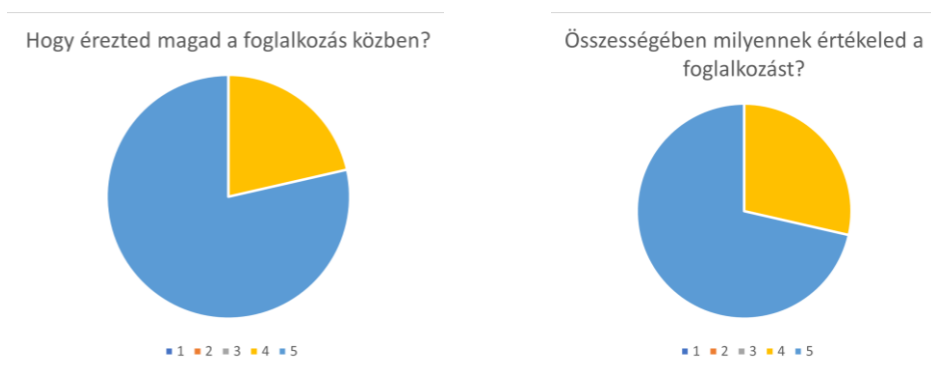
Az 5. ábrán látható diagram alapján a tanárszakos hallgatók jónak, kiválóan értékelték a foglalkozást. Örülünk a megerősítésnek, hogy általános és középiskolában is érdemes kipróbálni a módszert. Sajnos ez most kimaradt, de mindenképpen meg kellene ezt csinálni a téma teljessége kedvéért több szinten, több csoportban.

Már a 4-es jót jelent, azért nem lehet elmenni az utolsó kérdés eredményei mellett, ahol csupán hárman válaszoltak 5-össel; elképzelhető, hogy azért, mert már tisztában voltak a fogalmak nagyrészeivel, de természetesen az is lehet, hogy azt gondolják, nem ez a leghatékonyabb módja a megértésnek. Ezzel egyet is tudunk érteni. A módszertan szerintünk sem a leghatékonyabb, de a legszemléletesebb. A szoftverarchitektúrák vagy hálózati kommunikációk általános iskolában való oktatása egyáltalán nem triviális és ritkán valósul meg. A módszertan egy új alternatívát kínál erre a problémára.



**6. ábra:** Értékelj mennyire járult hozzá a foglalkozás, hogy közelebb kerülj/megértsd/szemléltesd az alábbi konkrét fogalmakat?

A 6. ábra eredményei alapján kijelenthető, hogy a hallgatók szerint a foglalkozás általában hozzájárult különböző informatikai fogalmak megértéséhez. A hálózati protokollokkal kapcsolatos kérdések különösen jól szerepeltek, ami azért lényeges, mert a foglalkozás alatt mi erre kívántuk helyezni a hangsúlyt. A „queue” fogalma maradt csak alul, ami érthető, mert éppen csak megemlítésre került, különösen nem beszéltünk róla.



7. ábra: Hogy érezted magad a foglalkozás közben?

8. ábra: Össességében milyennek értékeled a foglalkozást?

A 7. és 8. ábrán az látható, hogy a hallgatók összességében jól érezték magukat a foglalkozáson, és azt hasznosnak találták.

## 6. Konklúzió

Az informatikai gondolkodást számítógép nélkül is lehet oktatni és demonstrálni. A cikkben bemutattunk egy olyan szituációs játékot, melyet általános és középiskolás korosztályban is alkalmazni lehet, s a szoftverrendszerek, valamint a hálózati kommunikáció tervezésére helyezi a hangsúlyt. A gyakorlat azért is lehet hiánypótló, mert ezeket a fogalmakat nagyon nehéz demonstrálni.

A módszer magában hordoz módosítási lehetőségeket (melyek közül párat mi is bemutattunk), így a tanár egy-egy kiemelt fogalomhoz, a korosztályhoz és az előképzettséghez tudja igazítani a játékot.

## Irodalom

1. Tim Bell, Ian H. Witten, Mike Fellows: *CS Unplugged*  
[https://classic.csunplugged.org/documents/books/english/CSUnplugged\\_OS\\_2015\\_v3.1.pdf](https://classic.csunplugged.org/documents/books/english/CSUnplugged_OS_2015_v3.1.pdf) (utoljára megtekintve: 2021.11.19)
2. Bell, T., Witten, I.H., Fellows, M.: *Computer Science Unplugged: Off-Line Activities and Games for All Ages (Original Book)* (1999)
3. Erdősne Németh Ágnes: *Számítógép nélküli tevékenységek a számítástudomány alapjainak bemutatására és az algoritmusok tanításakor* In: Szlávi, Péter; Zsakó, László (szerk.) INFODIDACT 2017 (Budapest, Magyarország: Webdidaktika Alapítvány) – ISBN 978-615-80608-1-3, Paper: No 6, 10 p. (2017)  
<http://konferenciak.inf.elte.hu/infodidact/InfoDidact17/Manuscripts/ENA.pdf> (utoljára megtekintve: 2021.11.30.)
4. Duncan, C., Bell, T.: *A pilot computer science and programming course for primary school students*. In: WIPSCSE 2015, pp. 39–48 (2015)
5. Yvon Feastery, Luke Segarsz, Sally K. Wahbay, and Jason O. Hallstromy: *Teaching CS Unplugged in the High School (with Limited Success)*, ITiCSE 2011, Darmstadt
6. Erdősne Németh Ágnes: *Teaching Graphs for Contestants in Lower-Secondary-School-Age*. In: OLYMPIADS IN INFORMATICS 11: 1 pp. 41–53., 13 p. (2017)

7. Bernát Péter: *The Methods And Goals Of Teaching Sorting Algorithms* In: Public Education In ACTA DIDACTICA NAPOCENSIA 7: 2 pp. 1-10., 10 p. (2014)
8. Bell, T., Rosamond, F., Casey, N.: *Computer science unplugged and related projects in math and computer science popularization*. In: Bodlaender, H.L., Downey, R., Fomin, F.V., Marx, D. (eds.) *The Multivariate Algorithmic Revolution and Beyond: Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday*. LNCS, vol. 7370, pp. 398–456. Springer, Heidelberg (2012).
9. Clarke, B.: *Computer Science Teacher*. British Computer Society, Swindon (2017)
10. Hazzan, O., Lapidot, T., Ragonis, N.: *Guide to Teaching Computer Science: An Activity-Based Approach*. Springer, London (2011)
11. Code.org lesson: *Lesson 4: Dance Party: Unplugged*  
<https://curriculum.code.org/hoc/unplugged/4/> (utoljára megtekintve: 2021.11.19)
12. Bell T., Vahrenhold J. (2018) CS Unplugged—How Is It Used, and Does It Work?. In: Böckenhauer HJ., Komm D., Unger W. (eds) *Adventures Between Lower Bounds and Higher Altitudes. Lecture Notes in Computer Science*, vol 11011. Springer, Cham.  
[https://doi.org/10.1007/978-3-319-98355-4\\_29](https://doi.org/10.1007/978-3-319-98355-4_29)
13. Hassan W.H. *Current research on internet of things (iot) security: A survey*. *Comput. Netw*; 148:283–294 (2019)



# Elemi algoritmusok – prefix összegzés

Menyhárt László<sup>1</sup> Gábor, Zsakó László<sup>2</sup>  
(ORCID: 0000-0002-1574-4454, 0000-0002-4614-1509)

<sup>1</sup>menyhart@inf.elte.hu, <sup>2</sup>zsako@caesar.elte.hu  
ELTE Eötvös Loránd Tudományegyetem, Budapest  
Informatikai Kar

**Absztrakt.** Ebben a cikkünkben egy speciális problémát, a prefix összegzést (más néven kumulatív összegzést), és annak variációit mutatjuk be. Részletesen megvizsgáljuk az alproblémát és összehasonlítjuk a lehetséges megoldásait. Hallgatóink hasonló problémákat oldottak meg versenyeken, így a megoldásairól statisztikával rendelkezünk. A cikket olyan olvasóknak ajánljuk, akik érdeklődnek a programozási módszertan iránt vagy diákjaik versenyeken fognak indulni.

**Kulcsszavak:** módszeres programozás, oktatás, mérés, prefix összeg, kumulatív összeg.

## 1. Bevezetés

A programozás tanítás során az algoritmikus struktúrák (szekvencia, elágazás, ciklus) bevezetése után általában alapvető algoritmusokkal (a magyar szakirodalomban programozási tételekkel, típusalgoritmusokkal) foglalkoznak, mint pl. összegzés, keresés, maximumkiválasztás, rendezés stb.

A programozási versenyeken (és a tehetséggondozásban) megjelennek feladatmegoldási stratégiák (mint. pl. mohó stratégia, dinamikus programozás), a kettő közötti űrt azonban kevés helyen töltik ki. Ezt az űrt töltötte ki J. Bentley klasszikus cikksorozata és könyve (Programming Pearls [1], magyarul *A programozás gyöngyszemei* címen jelent meg) 1984-ből, újabban pedig például az algoritmizálást tanító weboldalak [2,3]. Az első megjelenése azonban 1954-es, egy nem közvetlen informatika témájú folyóiratban [4].

Ebben a cikkben azt járjuk körbe, hogy mihez kezdenek a kumulatív összegzés (prefix összegzés) feladattípussal magyar programozási versenyek résztvevői. Érdekességként két nagyon különböző korosztály vizsgálunk, az egyikben 13-16 éves diákok szerepelhetnek – Nemes Tihamér Online Programozási Verseny [5], a másikban pedig egyetemen informatikus szakra járó, a programozással most ismerkedő egyetemi hallgatók – Tehetségkutató Egyetemi Programozási Verseny [6].

Nem szándékozunk ebben a cikkben az algoritmus haladó alkalmazásaival foglalkozni (kiegészített bináris fákkal, Fenwick fákkal, ...).

## 2. Kumulatív összegzés alapfeladat

**Feladat** Adott egy  $N$  elemű pozitív és negatív számokat is tartalmazó számsorozat. Adjuk meg a sorozat azon  $[a, b]$  intervallumát, ahol az elemek összege maximális! Olyan  $a$  és  $b$  indexet kell megadnunk, amire az alábbi feltétel teljesül:

$$1 \leq a \leq b \leq N \quad \text{és} \quad \forall p, q (1 \leq p \leq q \leq N): \quad \sum_{i=a}^b X_i \geq \sum_{i=p}^q X_i$$

**A megoldás**

Az alapmegoldás kiszámolja minden intervallumra a számok összegét, majd ezekre maximumkiválasztást végez:

```

összeg(i, j) :
  S:=0
  Ciklus k=i-től j-ig
    S:=S+X[k]
  Ciklus vége
  összeg:=S
Függvény vége.

Alapmegoldás(max, a, b) :
  max:=-∞
  Ciklus i=1-től N-ig
    Ciklus j=i-től N-ig
      s:=összeg(i, j)
      Ha s>max akkor max:=s; a:=i; b:=j
    Ciklus vége
  Ciklus vége
Eljárás vége.

```

A három ciklus miatt a futási ideje  $N^3$ -bel arányos. Gyakran alkalmazható ötlet, hogy először nem azt számoljuk ki, amit a feladat kér, hanem valami egyszerűbbet, amiből viszont már könnyű megkapni a feladat megoldását.

Számítsuk ki az  $s$  vektorba a sorozat első  $i$  elemének összegét!

$$\forall i(0 \leq i \leq N): \quad s(i) = \sum_{j=1}^i x_j$$

ami rekurzív összefüggést használva, egyszerűbben:

$$\forall i(1 \leq i \leq N) \quad s(i) = s(i-1) + x(i), \quad s(0) = 0$$

Ezután egy intervallum elemei összege így fejezhető ki (1. ábra):

$$\sum_{i=a}^b x_i = s(b) - s(a-1)$$

Sorszámok	0	1	2	3	4	5	6	
Sorozat		3	5	1	8	2	4	
prefix sum	0	3	8	9	17	19	23	
sum(1,6)	0	3	8	9	17	19	23	
sum(1,2)	0	3	8	9	17	19	23	
sum(3,5)	0	3	8	9	17	19	23	19-8=11

1. ábra: prefix sum

Az alábbi algoritmus – a kumulatív összegzés módszere – így  $N^2$ -tel arányos lépést tesz meg.

Kumulatív összegzés (max, a, b) :

```
s[0]:=0
Ciklus i=1-től N-ig
  s[i]:=s[i-1]+X[i]
Ciklus vége
max:=-∞
Ciklus i=1-től N-ig
  Ciklus j=i-től N-ig
    Ha s[j]-s[i-1]>max akkor max:=s[j]-s[i-1]; a:=i; b:=j
  Ciklus vége
Ciklus vége
Eljárás vége.
```

### 3. Kumulatív összegzés variációk

Ez a stratégia nem csak összegzéskor (sorozatszámítások) használható, hanem más feladattípusoknál is. Alapvetően olyan esetekben, amikor a műveletek asszociatívak [7], van „inverze” [2], azaz

$$f(a, b) = f^{-1}(f(1, b), f(1, a - 1)).$$

Például

$$\sum_{i=a}^b X_i = \sum_{i=1}^b X_i - \sum_{i=1}^{a-1} X_i$$

vagy

$$X_a * X_{a+1} * \dots * X_b = \frac{X_1 * X_2 * \dots * X_b}{X_1 * X_2 * \dots * X_{a-1}}$$

feltéve, hogy egyik érték sem 0.

A maximumkiválasztás feladat  $[a, b]$  intervallumra azonban a fentiekre sem mindig érdekes, a szummánál például érdektelen, ha minden szám pozitív, a produktumnál pedig, ha minden szám nagyobb 1-nél.

Sokszor azonban a feladat nem maximumkiválasztás, hanem sok kérdés megválaszolása különböző intervallumokra [8,9]. Egy másik típusban bizonyos típusú szummát keresünk, a [10] versenyfeladatban például a leghosszabb sorozatot, ahol a számok összege osztható 7-tel.

A megszámlálás tételre is lehet értelme, de akkor vagy az intervallum hosszát, vagy valami más tulajdonságot kell módosítani.

Ilyenek lehetnek:

- adjuk meg egy sorozat legfeljebb  $K$  hosszú részét, amelyben az adott tulajdonságú elemek száma maximális;
- adjuk meg egy sorozat legfeljebb  $K$  hosszú részét, amelyben az adott tulajdonságú elemek száma maximális és a két szélső elem is adott tulajdonságú;
- adjuk meg egy sorozat leghosszabb részét, amelyben az elemek legalább fele adott tulajdonságú és a két szélső elem!

Sok feladatban a módszer kiegészítőként szerepel, ráadásul csak bizonyos előfeltételek teljesülése esetén.

**Feladat:** Egy véletlenszám generátor 0 és  $M-1$  közötti számokat állít elő. Megkaptuk a generátor által előállított első  $N$  darab véletlenszámot. Írj programot, amely a generátor ellenőrzéséhez kiszámolja a sorozat  $K$  darab  $[A, B]$  intervallumára, hogy hány 0 és  $M-1$  közötti szám nem fordul elő az adott intervallumban!  $K$  értéke sokkal nagyobb, mint  $M$  értéke!

Írhatunk rá egy nem annyira naiv megoldást (indexeléssel oldja meg a számlálást):

```
Nincs (N, X, K) :
  Ciklus i=1-től K-ig
    Be: A, B
    szam[] := (0, ..., 0)
    Ciklus j=A-től B-ig
      szam[X[j]] := szam[X[j]] + 1
    Ciklus vége
  C := 0
  Ciklus j=0-től M-1-ig
    Ha szam[j]=0 akkor C:=C+1
  Ciklus vége
  Ki: C
Ciklus vége
Eljárás vége.
```

A futási ideje ezek szerint  $K \cdot (N+M)$ -mel arányos.

A kumulatív összegzés először kiszámolja egy mátrixban, hogy az első  $i$  véletlenszámból hány  $j$  érték van, majd ebből számolja az eredményt:

```
Nincs (N, X, K) :
  szam[0] := (0, ..., 0)
  Ciklus i=1-től N-ig
    szam[i] := szam[i-1]; szam[i, X[i]] := szam[i, X[i]] + 1
  Ciklus vége
  Ciklus i=1-től K-g
    Be: A, B
    C := 0
    Ciklus j=0-től M-1-ig
      Ha szam[b, j] - szam[a-1, j] = 0 akkor C := C+1
    Ciklus vége
  Ki: C
  Ciklus vége
Eljárás vége.
```

Az első ciklus a tömbértékadás lassú megvalósítása esetén is  $N \cdot M$ -es, a második pedig  $K \cdot M$ -es. Mivel a feladatleírás szerint  $K$  sokkal nagyobb  $M$ -nél, azért  $K \cdot N + K \cdot M$ , akkor nagyobb  $N \cdot M + K \cdot M$ -nél, és ez akkor igaz, ha  $K > M$ , mint a feladatleírásban szerepelt.

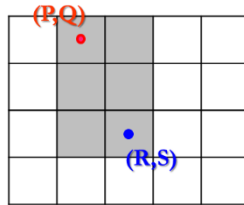
## 4. Kumulatív összegzés és matematika

A feladat nemcsak vektorokra, hanem más struktúrákra is általánosítható, például a következőben egy mátrixra [11]:

**Feladat:** Egy földműves egy téglalap alakú területet szeretne vásárolni egy  $N \times M$ -es téglalap alakú földterületen. Tudja minden megvásárolható földdarabról, hogy azt megművelve mennyi lenne a haszna vagy vesztesége. Add meg azt a téglalapot, amelyen a legnagyobb haszon érhető el!



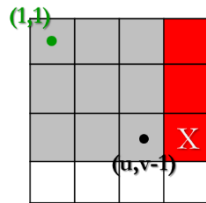
Az elemi megoldásban minden  $(P, Q)$  bal felső,  $(R, S)$  jobb alsó sarkú résztéglalapnak ki kellene számolni az elemei összegét, ami 6 ciklus eredményezne, majd ezek maximumát vehetnénk (2. ábra).



2. ábra: feladat

Próbáljunk valami részcélt kitűzni: számoljuk ki az  $(1, 1)$  bal felső,  $(u, v)$  jobb alsó sarkú téglalapok értékét!

$X =$  a szürke téglalap értéke + a piros téglalap összege (3. ábra)

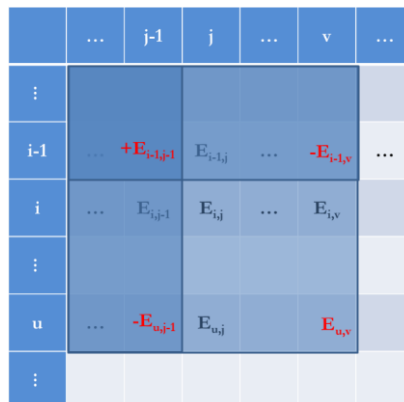


3. ábra: E kiszámítása

```

E[1..N, 0] := (0, ... 0)
Ciklus v=1-től M-ig
  x:=0
  Ciklus u=1-től N-ig
    x:=x+T[u, v]
  Ciklus vége
E[u, v] := E[u, v-1] + x
Ciklus vége
    
```

Az  $(i, j)$  bal felső,  $(u, v)$  jobb alsó sarkú téglalap értékei összege ezek alapján: érték  $(E, i, j, u, v) = E[u, v] - E[u, j-1] - E[i-1, v] + E[i-1, j-1]$  (4. ábra)



4. ábra: az eredmény kiszámítása

Az algoritmust csak azért írjuk le, hogy a következő lépésben módosíthassuk:

```

max:=-∞
Ciklus i=1-től N-ig
  Ciklus j=1-től M-ig
    Ciklus k=i-től N-ig
      Ciklus l=j-től M-ig
        o:=érték(E, i, j, k, l)
        Ha o>max akkor P, Q, R, S:=i, j, k, l; max:=o
      Ciklus vége
    Ciklus vége
  Ciklus vége
Ciklus vége

```

Jól látható a 4 ciklus, azaz ha  $N$  és  $M$  azonos nagyságrendű, akkor a futási idő  $O(N^4)$ , azaz  $n$  negyedik hatványával arányos.

A versenyeken mindkét korosztálynak ugyanúgy módosítottuk a feladatot, rögzítettük a kiválasztandó téglalap területét, azaz olyan négy értéket kerestünk, amelyre egy újabb feltétel volt, hogy  $(R-P+1) * (S-Q+1) = A$ , egy adott  $A$  konstansra, illetve összeg helyett valamilyen típusú elemek számát kellett megadni.

A 2017/18 évben megrendezett verseny 3. fordulójának egyik feladata így szólt: „Egy téglalap alakú területen ismerjük  $N * M$  pont tengerszint feletti magasságát. Csúcsnak nevezzük azokat a pontokat, amelyek nagyobbak négy szomszédjuknál. A terület szélein biztos nincs csúcs. Készíts programot, amely megad egy pontosan  $A$  pontot tartalmazó téglalap alakú területet, amelyen a lehető legtöbb csúcs van!”

A versenyzők jelentős része, ahogy a korábbiak alapján várható volt, a naiv megoldást adta, 6 ciklussal. A módszert ismerők (kitalálók) jelentős része a fenti 4 ciklus belsejében vizsgálta a terület méretét:

```

Ha (k-i+1) * (l-j+1) = A akkor
  o:=érték(E, i, j, k, l)
  Ha o>max akkor P, Q, R, S:=i, j, k, l; max:=o

```

Az ennél picit ügyesebbek még egy ciklust megspórolhattak volna és az  $l$  változó értékét csak olyan  $k$  értékekre számolták ki, amelyekre  $(k-i+1)$  osztotta  $A$ -t, de ilyen megoldást alig találtunk.

```

max:=-∞
Ciklus i=1-től N-ig
  Ciklus j=1-től M-ig
    Ciklus k=i-től N-ig
      Ha A mod (k-i+1) = 0 akkor
        o:=érték(E, i, j, k, j+A div k-1)
        Ha o>max akkor P, Q, R, S:=i, j, k, j+A div k-1; max:=o
      Ciklus vége
    Ciklus vége
  Ciklus vége

```

Az igazi megoldásban hatékonyan, négyzetgyök( $A$ )<sup>1</sup> lépésben előre kiszámolhatók az  $A$  osztói, egy  $C$  elemű  $D$  vektorba, hiszen a terület két oldal szorzataként számolható.

<sup>1</sup> Elég az osztókat négyzetgyök( $A$ )-ig meghatározni, hiszen ha  $x$  osztója  $A$ -nak, akkor  $A/x$  is osztója.

Így a várt megoldás  $O(N^2C)$ , kihasználva, hogy  $D[k] * D[C-k+1] = A$ :

```

max:=-∞
Ciklus i=1-től N-ig
  Ciklus j=1-től M-ig
    Ciklus k=1-től C-ig
      o:=érték(E, i, j, i+D[k]-1, j+D[C-k+1]-1)
      Ha o>max akkor P,Q,R,S:=i, j, i+D[k]-1, j+D[C-k+1]-1; max:=o
    Ciklus vége
  Ciklus vége
Ciklus vége
    
```

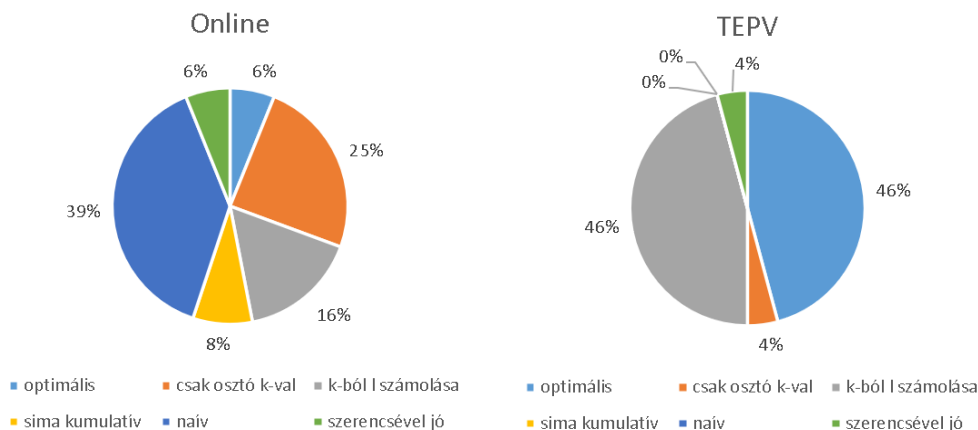
A tesztesetek alapján elértük, hogy a különböző megoldási módok eltérő pontszámot érjenek el, így a versenyzők által elért pontszámok alapján meghatározhatjuk megoldási típusok eloszlását (5. ábra).

Az egyes csoportokon belüli eltérő pontszám a speciális esetek jó vagy rossz kezelése miatt lehet.

optimális, csak osztó k-val, k-ból l számolása	92-100
csak osztó k-val, csak osztó l-lel	71-91
k-ból l számolása	55-70
sima kumulatív	41-54
naiv	16-40
szerencsével jó	0-15

5. ábra: megoldási módok ponthatárai

Látszik, hogy az egyetemisták a több matematikai ismeretüket felhasználva több pontot érnek el, illetve a naiv és a kumulatív megoldásból már tovább léptnek egy hatékonyabb megoldás felé. Náluk naiv és sima kumulatív megoldás nem is volt. Ezzel szemben ezek együtt a 13-16 éves korosztálynál majdnem elérték az 50%-os arányt. A legjobbak azonban közülük is megtalálták az optimális megoldást (6. ábra).



6. ábra: A megoldástípusok eloszlása

## 6. Konklúzió

A kumulatív összegzés alapgondolata tehát: Tetszőleges részsorozaton számolt érték helyett csak olyan részsorozatokra számoljunk értéket, amelyek kezdete az egész sorozat kezdete, majd a részsorozaton számolt értéket fejezzük ki ezekből!

Összegzés helyett megszámlálási feladatokra is alkalmazható ugyanez az elv, azaz a művelet bármely asszociatív művelet lehet, amelynek számolható az inverze.

A verseny tapasztalataink alapján a diákok az alpmódszert könnyen megtanulják, de maguktól nem találják ki – jellemzően minél fiatalabbakat nézünk, annál inkább nő a naiv, lassú megoldást készítő aránya. A kicsi módosításokat csak egy részük veszi észre, pl. amikor előlről és hátulról is kell számolni ilyen összegeket, vagy összegzés helyett más műveletet kell használni.

Fontos gondolat, hogy a naiv megoldások a legtöbb feladat esetén összegzést és maximumkiválasztást tartalmaznak, de a hatékony, kumulatív összegzést alkalmazók megírásához is ezek az elemi algoritmusok (programozási tételek) kelljenek. Azaz nem új algoritmusokról van szó, hanem a megszokott algoritmusok alkalmazásában kell másképp gondolkodnunk!

A matematikai megfontolások, mint az utolsó feladatban láthattuk, azonban a matematikából nem erős diákoknak problémát okoznak. A matematika pedig csak annyi volt, hogy ha egy egész szám két egész szám szorzataként áll elő, akkor mindkét számnak osztania kell a szorzatot – tehát szükség van az osztók előállítására. Az eredményekből is látható, hogy az első félévre járó egyetemisták lényegesen nagyobb arányban oldották meg a feladatot maximális pontszámmal.

A fentiek miatt úgy gondoljuk, hogy a programozás oktatásában a fenti alpmódszerrel (minden bonyolultabb adatstruktúrát kerülve) mindenképpen foglalkoznunk kell. Ebben a tanárok jelenleg kevés segítséget kapnak [12], mert a tudományos cikkek elsőprő többsége a különböző fákat használó fejlett megoldásokról szól.

## Irodalom

1. J. Bentley: *Programming Pearls - Algorithm Design Techniques*. In: Comm. ACM (1984), Vol 27. Issue 9, pp. 865-871
2. Johan Sannemo: *Principles of Algorithmic Problem Solving*. <https://usaco.guide/PAPS.pdf#page=207>, 2018 (utoljára megtekintve: 2021.11.15.)
3. ACM; <https://helloacm.com/category/teaching-kids-programming/> (utoljára megtekintve: 2021.11.15.)
4. E. S. Page: *Continuous Inspection Schemes. Biometrika*. Vol. 41. No. 1/2 (Jun., 1954), pp. 100-115, [https://www.jstor.org/stable/2333009?origin=JSTOR-pdf&seq=1#metadata\\_info\\_tab\\_contents](https://www.jstor.org/stable/2333009?origin=JSTOR-pdf&seq=1#metadata_info_tab_contents) (utoljára megtekintve: 2021.11.15.)
5. Nemes Tihamér Online Programozási Verseny; <http://tehetseg.inf.elte.hu/nemes-online/index.html> (utoljára megtekintve: 2021.11.15.)
6. Tehetségkutató Egyetemi Programozási Verseny; <http://versenyek.inf.elte.hu/versenyek/tehetsegkutato-egyetemi-programozasi-verseny/> (utoljára megtekintve: 2021.11.15.)
7. Guy E. Blelloch: *Prefix Sums and Their Applications. School of Computer Science*, Carnegie Mellon University, Pittsburgh, <https://www.cs.cmu.edu/~guyb/papers/Ble93.pdf> (utoljára megtekintve: 2021.11.15.)
8. Darren Yao, Dustin Miaoh: *Introduction to Prefix Sums*. <https://usaco.guide/silver/prefix-sums?lang=cpp> (utoljára megtekintve: 2021.11.15.)
9. USACO 2015 December Contest, Silver, *Problem 3. Breed Counting*. <http://www.usaco.org/index.php?page=viewproblem2&cpid=572> (utoljára megtekintve: 2021.11.15.)

10. USACO 2016 January Contest, Silver Problem 2. Subsequences Summing to Sevens.  
<http://www.usaco.org/index.php?page=viewproblem2&cpid=595> (utoljára megtekintve: 2021.11.15.)
11. Steven Halim: Classical problems.  
<https://acm.wustl.edu/cse232/classical.md> (utoljára megtekintve: 2021.11.15.)
12. Szlávi P., Zsakó L.: Elemi algoritmusok (a programozási tételek után). ELTE Informatikai Kar, Budapest.  
[http://tehetseg.inf.elte.hu/szakkorefop2017/pdf/elteikszakkor\\_elemi\\_algoritmusok.pdf](http://tehetseg.inf.elte.hu/szakkorefop2017/pdf/elteikszakkor_elemi_algoritmusok.pdf) (utoljára megtekintve: 2021.11.15.)



# Online programozás szakkörök – a tehetség gondozás jövője?

Nikházy László

nikhazy@inf.elte.hu

ELTE IK

**Absztrakt.** Az utóbbi években online térbe kényszerült oktatás a sok hátulütője mellett új perspektívákat is nyitott. A virtuális világban ugyanis nagyon könnyen és hatékonyan lehet az ország különböző pontjairól csatlakozó diákoknak foglalkozást szervezni. Mivel a tehetséges gyerekek alapvetően elszórta tünnek fel, egy-egy „hagyományos” iskolai tehetség gondozó szakkörön jellemzően vagy nagyon kevesen vannak, vagy nagy tudásbeli különbségek fordulhatnak elő. Kiváltképp igaz ez programozásból, ahol a tanulmányi versenyek anyaga jóval túlmutat az iskolai törzsanyagban. Az online szakkörökön a sok különböző helyen élő, azonos érdeklődésű és hasonló tudású diákokból tudunk csoportokat alkotni, ahol közösségben tanulhatnak a nekik legmegfelelőbb szinten. A résztvevő diákok körében végeztünk felmérést ennek a lehetőségnek az értékeléséről.

**Kulcsszavak:** programozás tehetség gondozás, informatika szakkör, online oktatás.

## 1. Bevezetés

A COVID-19 járvány miatti sok iskolabezárás és online oktatás világszerte negatívan érintette a gyerekek mindennapi életét és fejlődését [1]. A szociális kapcsolatok hiánya, az otthoni bezártság és a csupán virtuális, ingerszegényebb tanulási környezet a fiatalok mentális egészségét is veszélyeztetheti [2, 3]. A megváltozott oktatási forma kihát a diákok teljesítményére is. Clark és társai [4] azt figyelték meg, hogy általában javultak a diákok pontszámai és jegyei az online oktatás közben, kivéve a legjobb tanulóké. Különösen sokat tudtak fejlődni az olyan, korábban gyengén teljesítő diákok, akik az iskolájukon kívüli, magasabb kvalitású oktatók által tartott órákon vettek részt virtuálisan.

A sok nehézség és kihívás [5] mellett az online oktatásban szerzett tapasztalatok a diákok és a tanárok részéről is új lehetőségeknek adnak utat [6]. 2020 tavaszán és a 2020-2021. tanévben a programozás tehetség gondozó szakkörököt és versenyre felkészítő foglalkozásokat is csak az interneten tarthattuk meg. Így viszont az ország bármely pontjáról, és határon túlról is könnyen tudtak csatlakozni diákok. A közoktatásban megszerzett rutin, a kialakított technológiai megoldások, a mindenkinek meglévő otthoni számítógép és internetkapcsolat révén ma már szinte teljesen akadálymentes ezeknek az online foglalkozásoknak a megszervezése és lebonyolítása.

A tehetség gondozásban hatalmas előny az, ha megszűnnek a földrajzi korlátozások, és ezáltal az egymástól távol élő, kiemelkedő képességű gyerekekből tudunk csoportokat alkotni. Hiszen a tehetséges gyerekek alapvetően elszórta tünnek fel, és sok helyen nincs megfelelő pedagógus, vagy nincs nekik való foglalkozás, amin fejlőhetnek. Amennyiben van is egy-egy iskolában programozás szakkör, valószínűleg nagy skálán mozog a gyerekek tudása és képessége, ezért nehezen valószínűsíthető meg, hogy mindenki fejlődjön. Nem segít az sem, hogy kevés diák foglalkozik programozással, sok iskolában nem gyűlik össze egy csoportnyi gyerek sem, akikkel egy egészséges létszámú közösséget lehetne kialakítani.

Az országos szinten rendezett online tehetség gondozó szakkörökhöz elég egy-két kiemelkedő tanár, akik központilag tarthatják a foglalkozásokat. A kivételes képességű diákokból tudunk olyan

csoportokat csinálni, amelyekben hasonló tudásszinten lévő gyerekek vannak, ennek hasznáról és lehetőségeiről Erdősne is hosszan ír [7]. Az egész országban vannak annyian a programozásban kiemelkedő tanulók, hogy 10 és 20 közötti létszámú csoportokban mindenki tudjon fejlődni a saját szintjéhez képest. Egy további előnye az online óráknak, hogy időpontot is könnyebb találni, mivel hétvégén és hétköznap késő délután is lehet őket tartani, nem csak az iskolai órák után. Természetesen megvannak a maga hátrányai a virtuális térben tartott foglalkozásoknak, hiszen fellelphetnek technikai problémák vagy nehezítő körülmények, hiányoznak a személyes kapcsolatok és sokkal kevesebb visszacsatolást kapnak a tanárok és a diákok is.

Mindent egybevéve úgy gondoljuk, hogy az előnyök jócskán túlszárnyalják a hátrányokat, és az online tehetséggondozó szakkörök hiánypótlóak, mivel sok diák számára egy teljesen új lehetőséget, vagy a korábbiakhoz képest sokkal jobb lehetőséget nyújtanak. Ez az első hipotézisünk, aminek igazolására végeztük a jelenlegi kutatást. A második hipotézisünk ennek hátterére vonatkozik. Feltételezésünk szerint a szakkörök elsődleges újdonsága az, hogy megfelelő szintű csoportos oktatást tudunk nyújtani a kiemelkedően tehetséges diákoknak is. A szakköreinken résztvevő diákok által kitöltött kérdőívek segítségével gyűjtöttünk adatokat a hipotézisek vizsgálatára. A 2. fejezet röviden bemutatja a szakköröket, a 3. fejezet a felhasznált kérdőívet írja le, míg a 4. fejezet a kapott eredményeket és az elemzésüket tartalmazza.

## 2. A szakkörök

A vizsgált online szakkörök a nemrég létrehozott Alphacademy [8] online magániskola szervezésében valósulnak meg, ahol a szerző és Weisz Ágoston mellett számos korábbi tanítványuk is vezet csoportokat. Az Alphacademyben teljesen kezdő szinttől a nemzetközi versenyfelkészítésig tanítunk programozást gondolkodtató módon, a fiatalabbaknak Logo, az idősebbeknek, illetve haladóbbaknak C++ nyelven. A diákok a tudásuk szerint vannak csoportokba osztva, a belépéskor megoldott szintfelmérő feladatok alapján, illetve később az órákon mutatott teljesítményük alapján. A csoportok létszáma 10 és 20 fő közötti.

A szakkörökön a tanulás alapvetően problémamegoldáson keresztül valósul meg [9]. Egy-egy új téma tanításakor törekszünk arra, hogy először feladatokat adjunk, melyek megoldása során a diákok felfedezhetik a szükséges módszereket. Különösen fontosnak tartjuk, hogy időt adjunk mindenkinek arra, hogy egyénileg vagy csoportosan gondolkodjanak a feladatokon. A megoldás megbeszélése közben tudatosítjuk, hogy milyen módszereket tanultunk ennek kapcsán. Amennyiben egy-egy összetettebb algoritmust vagy adatszerkezetet bemutatunk nekik, azt különböző feladatokban, több újfajta helyzetben kell alkalmazniuk. Korábbi tanítási tapasztalataink alapján hangsúlyt fektetünk arra, hogy a megbeszélte feladatokra programot is írjunk, amelyet online kiértékelővel le tudunk tesztelni, még ha ez sok időt el is visz a foglalkozásokból.

Az online oktatás színtere a Discord [10] platform, amelyet korábban online programozás táborok tartására is alkalmaztunk és nagyon pozitív visszajelzéseket kaptunk [11]. Minden csoportnak kéthetente van 90 perces online órája. A foglalkozások között aktív egyéni munkát várunk el, és segítséget is nyújtunk. Az óra előtt legalább egy héttel kiadunk a diákoknak 3-5 feladatot, így bőven van idejük ezeken gondolkodni. Ezek egy része ismétlődő feladat, amely a korábbiak elmélyítésére irányul, egy része pedig az új anyag felvezetése, tehát felfedeztetés a cél. Adunk hozzájuk néhány egymondatos segítséget is, amelyeket önkiszolgáló módon nézhetnek meg. Az órán újabb 3-5 feladatot adunk fel, amiken 2-3 fős csoportokban dolgoznak a diákok. A csoportmunkában ösztönözzük őket, hogy dolgozzanak össze, segítsenek egymásnak. A feladatok elméleti megoldását közösen találják ki, a programozásra általában felkínáljuk azt a két lehetőséget, hogy 1) közösen csinálják: egyikük írja, a másik figyel és segít („pairwise programming”), vagy 2) külön-külön megcsinálják, majd megnézik egymásét és kijavítják a hibákat. A csoportok privát virtuális „szobában” dolgoznak



(hang- illetve videókapcsolattal), mialatt mi, tanárok körbe járunk a szobákban, figyeljük a haladásukat és segítségeket adunk.

Ebben a felmérésben a haladó csoportokat vizsgáltuk csak, vagyis azokat, ahol már nem a programozási alapokat, hanem problémamegoldási stratégiákat és algoritmusokat tanítunk. Ha a hazai versenyek szintje alapján szeretnénk meghatározni, akkor nagyjából a Nemes Tihamér Verseny [12] 1. korcsoport döntős szintjétől az Olimpiai Válogatóverseny [13] szintjéig vannak különböző csoportjaink, egészen pontosan 5 csoportról van szó. Az egyes csoportok létszámai: 15, 14, 19, 18 és 19 fő. Egy-egy csoportban különböző korú, hasonló tudású diákok vannak. A csoportok tanárai Csernán András, Szabó Kornél, Tóth Gellért és a szerző (2 csoport).

### 3. Felmérés

A felmérést az 5 haladó szakköri csoportba járó diákok körében végeztük, a kitöltés önkéntes és anonim volt. A csoportok összlétszáma 85, közülük 37 gyerek töltötte ki a kérdőívet. Feltettünk néhány arra vonatkozó kérdést, hogyan viszonyítják az online programozás szakköröket más lehetőségekhez:

- Hiánypótlónak érzed az online programozás szakköröket?
- Az online szakkörökön mennyire lehet hatékonyan tanulni, jelenléti foglalkozásokhoz hasonlítva?
- Az online szakkörökön mennyire lehet kényelmesen tanulni, jelenléti foglalkozásokhoz hasonlítva?
- Szerinted milyen előnye van az online szakköröknek?
- Szerinted milyen hátránya van az online szakköröknek?

Mivel a helyi programozás szakkörökkel és informatika órai programozás oktatással összevetve értékeljük az online szakköröket, ezért megkérdeztük, hogy részt vesznek/vettek-e ilyeneken. Arról is érdeklődtünk, hogy hányan járnak a szakkörökre és egyáltalán hányan vannak az iskolában, akik hasonló szinten programoznak:

- Mit gondolsz, az iskolában kb. hányan vannak még, akik hozzád hasonló szinten foglalkoznak programozással?
- Van-e, vagy volt-e az utóbbi 1-2 évben az iskolában, vagy a városban programozás szakkör, amire jártál?
- Nagyjából hányan vannak az iskolai szakkörön?
- Van-e, vagy volt-e az utóbbi 1-2 évben az iskolai informatika órán programozás oktatás?

Mindhárom típusú oktatásról feltettük ugyanazokat a kérdéseket, amelyek a hipotézisünk alátámasztása szempontjából érdekesek. Tehát a foglalkozások és a résztvevők szintje, a szakmai színvonal és a személyes fejlődés volt ezeknek a kérdéseknek a témája:

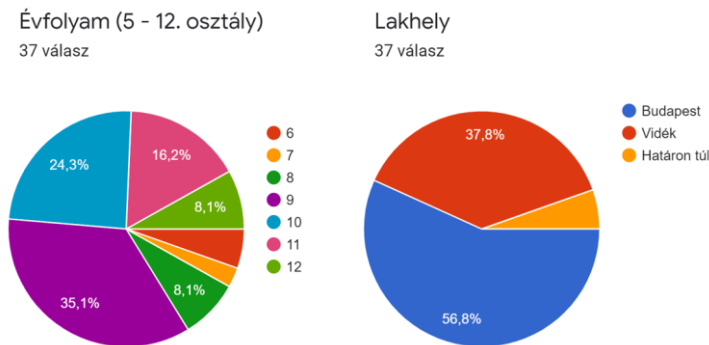
- Összességében milyenek értékeled az [online programozás szakköröket / iskolai programozás szakköröket / informatika órakon a programozás oktatást]?
- Milyen az [online szakkörök / iskolai szakkörök / informatika órakon tanult programozás] szakmai színvonala? (Értsd: nívós programozási tudást lehet megszerezni?)
- A tudásszintednek megfelelő oktatásban részesülsz? Jó neked [az online szakkör / iskolai szakkör / informatika órai programozás] szintje?
- Megítélésed szerint a társaid milyen szinten vannak hozzád képest?

- Az [online szakkörökön / iskolai szakkörökön / informatika órákon] érdemben tudsz fejlődni?
- Milyenek tartod az [online szakkörökön / iskolai szakkörökön / informatika órákon] a közösséget?

## 4. Eredmények

### 4.1. A kitöltők és az iskolai programozás oktatásra vonatkozó adatok

A kérdőívet kitöltő 37 diák évfolyam és lakhely szerinti eloszlása látható az alábbi ábrán. A kitöltők mintegy háromnegyede 9-11. osztályos tanuló, de vannak közöttük 6-12. osztályig minden évfolyamon. Különösen fontos számunkra, hogy a válaszadók majdnem fele vidéki vagy határon túli diák, egyes esetekben az ő válaszaikat külön is elemezzük. Ennél részletesebb adatokat a kérdőív anonimitása miatt nem kérdeztünk. A diákok mintája nem biztos, hogy reprezentatív, és viszonylag kis méretű, de elegendő mennyiségű ahhoz, hogy releváns következtetéseket vonjunk le belőle.

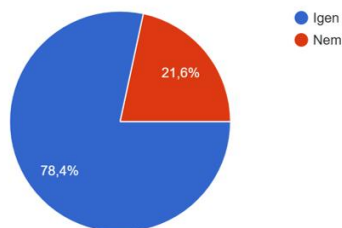


1. ábra: Évfolyam és lakhely szerinti eloszlás.

Üdvözítő az a tény, hogy a kitöltő diákok mintegy 80%-a tanult programozni az iskolai informatika órán is. (Valószínűleg, ha az ország összes diákját vizsgálnánk, akkor nem ilyen arányt kapnánk, de nincs erre vonatkozó adatunk.) Egy későbbi ábrán látható lesz, hogy az informatika óra programozás színvonala nem nekik megfelelő, ami egyáltalán nem meglepő, hiszen kiemelten tehetséges diákokról van szó. Szomorú viszont az alábbi ábrán jobb oldalon látható kördiagram, miszerint az esetek 35%-ában nem jártak szakkörre az iskolájukban vagy a városban. Tehát a továbbiakban, amikor arról értekezünk, hogy az online szakkör és az iskolai szakkör mennyire megfelelő a diákoknak, emlékezzünk arra is, hogy több, mint az egyharmaduknál nem is beszélhetünk arról, hogy megfelelő-e, mert nincs.

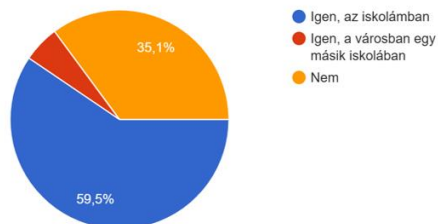
Van-e, vagy volt-e az utóbbi 1-2 évben az iskolai informatika órán PROGRAMOZÁS oktatás?

37 válasz



Van-e, vagy volt-e az utóbbi 1-2 évben az iskolában, vagy a városban programozás szakkör, amire jártál?

37 válasz

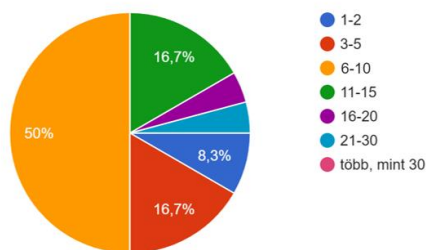


2. ábra: Informatika órán programozás oktatás, és iskolai szakkörök.

Megkérdeztük a gyerekektől, hogy hányan járnak az iskolai szakkörökre, és megítélésük szerint az iskolájukban hányan vannak, akik hozzájuk hasonló szinten foglalkoznak programozással. A felmérésben résztvevők döntő többsége alacsony létszámú szakkörre jár. A válaszok 75%-ában a szakköri létszám legfeljebb 10, ezen belül a válaszok 25%-ában a létszám nem több 5-nél. Összevetésképpen, az online szakköri csoportok létszáma 14 és 19 között mozog. Szomorú adat, hogy a diákok mintegy 60%-a szerint az iskolájukban legfeljebb 5 olyan diák van, aki hozzá hasonló szinten van programozásból. Ha ez így van, akkor gyakorlatilag képtelenség szervezni szakköri csoportokat megfelelő szinten, csupán egyéni tehetséggondozó foglalkozásoknak van értelme. Érdekes, hogy az online szakkör előnyeiről szóló szabad szöveges válaszokban többen megfogalmazták pozitívumként, hogy ott magasabb a létszám: „Több ember részt tud venni rajta, nincs az a probléma, hogy sokan nem ugyan ott élnek.”; „Így nem csak egy város diákjainak lehet a szakkört megtartani.”; „Magasabb részvételi arány”.

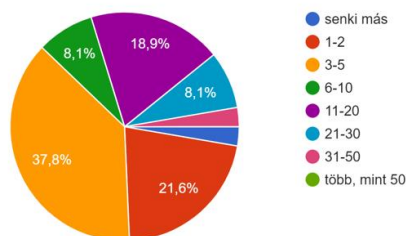
Nagyjából hányan vannak az iskolai szakkörön?

24 válasz



Mit gondolsz, az iskolában kb. hányan vannak még, akik HOZZÁD HASONLÓ SZINTEN foglalkoznak programozással?

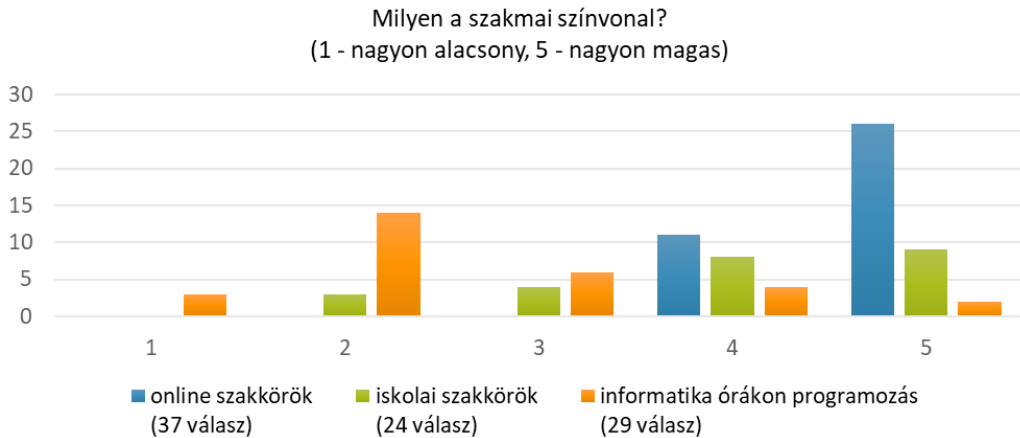
37 válasz



3. ábra: A szakkörök létszáma és a hasonló szinten lévő iskolatársak száma.

## 4.2. A háromféle programozás oktatás értékelése

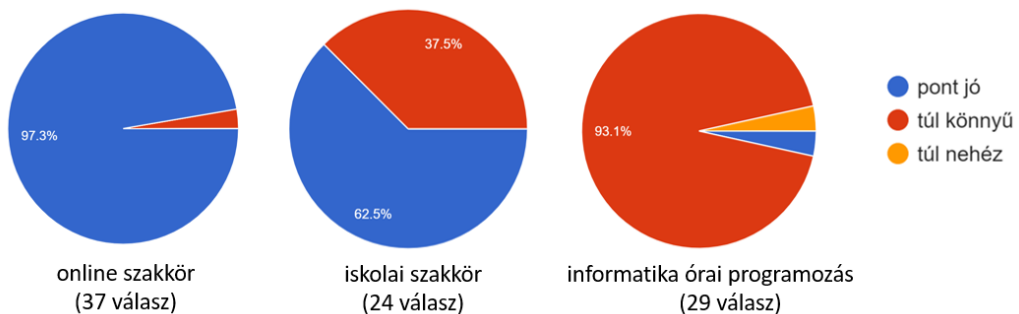
Az online szakkörök szakmai színvonalát (nívós-e a programozás oktatás) jelentősen magasabbra értékelték a diákok, mint az iskolai szakkörök, illetve az informatika órák esetében ugyanezt (átlagok sorrendben: 4,70; 3,96; 2,59). Az eloszlások az alábbi oszlopdiagramon láthatók. A különbség természetesen adódhat a tanár, a tananyag és a tanítási módszerek eltéréséből is, de meglátásunk szerint nagy szerepe van ebben a foglalkozáson résztvevőknek, hiszen az oktató alkalmazkodik hozzájuk. A továbbiakban elemzett válaszok alátámasztják ezt.



4. ábra: Szakmai színvonal értékelése.

Kiemelt fontosságú a szempontunkból az a kérdés, hogy a diákok a tudásszintjüknek megfelelő oktatásban részesülnek-e. Nagyon erős indikátor, hogy az online szakkörökről 97%-ban mondták azt, hogy pont jó a szakkör szintje. Az iskolai szakkörök esetében ez a szám csak 62,5%, vagyis a diákok 37,5%-a még a szakkört is túl könnyűnek tartja. Ne feledkezzünk meg arról, hogy iskolai szakkör nincs mindenhol, az előbb említett 62,5% az összes kitöltőnek csak 40%-a, ami azt jelenti, hogy az online szakkörökre járó diákok kevesebb, mint felének van a tudásszintjének megfelelő jelenléti szakköre. Tovább elemezve ezeket a válaszokat kiderül, hogy a többségük budapesti diák, mindössze 4 vidéki vagy határon túli diák van, aki ezt állította, ami a vidéki / határon túli kitöltőknek a 25%-a. Tehát az online szakkörökre jelentkező, Budapesten kívül élő diákok háromnegyede nem tud részt venni az iskolájában vagy a városában neki megfelelő szintű programozás szakkörön, míg budapesti diákoknál ez az arány kicsit kevesebb, mint a válaszadók fele. Az informatika órai programozás oktatás szintje a kutatásban résztvevő tehetséges gyerekeknek túl könnyű (93%), de ez nem meglepő adat, bár ideális esetben részesülhetnének differenciált oktatásban.

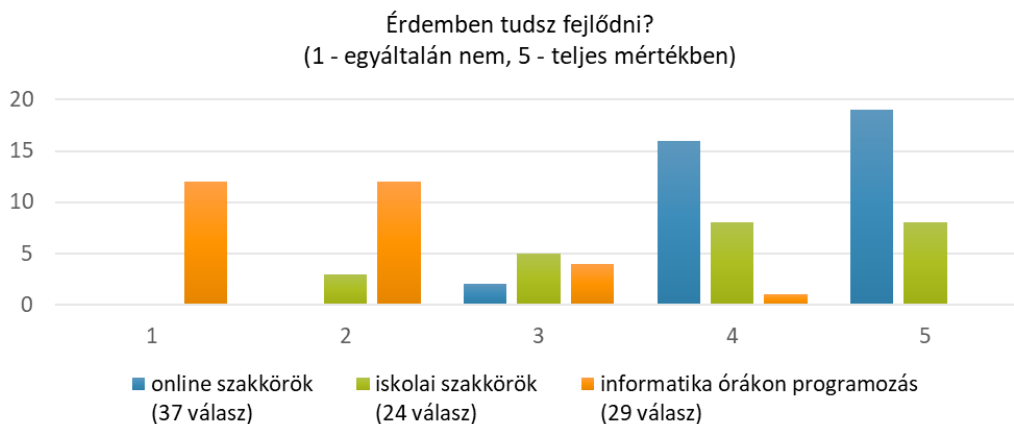
A tudásszintednek megfelelő oktatásban részesülsz? Jó neked a ... szintje?



5. ábra: Az oktatás szintjének viszonya az egyéni tudásszinthez.

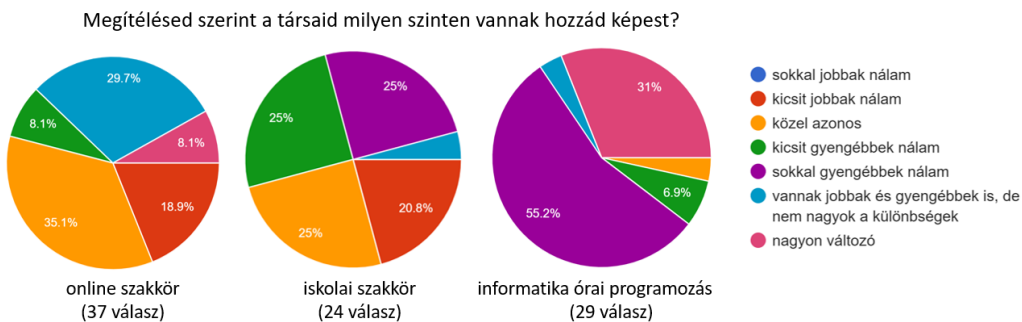
Ehhez kapcsolódik az a kérdés, hogy tudnak-e a diákok érdemben fejlődni a foglalkozáson. Az alábbi ábra mutatja a kapott válaszok eloszlását a három különböző foglalkozásra. Az eloszlásokból

látható, hogy az online szakkörök esetén valósul meg ez leginkább (átlag: 4,46), de az iskolai szakkörök (átlag: 3,88) is messze az informatika órák felett vannak (átlag: 1,79).



6. ábra: Érdemi fejlődés értékelése.

Amikor arról kérdeztük a résztvevőket, hogy a társaik milyen szinten vannak hozzájuk képest, az iskolai szakkörökről az esetek felében nyilatkoztak úgy, hogy a többiek kicsit vagy sokkal gyengébbek náluk, míg az online szakkörök esetében csupán 8% mondta azt, hogy a többiek kicsit gyengébbek náluk. Senki nem nyilatkozott úgy, hogy a többiek sokkal gyengébbek nála. 35% érzi úgy, hogy közel azonos szinten van mindenki a csoportban, és további 30% nyilatkozta, hogy vannak jobbak és gyengébbek is nála, de nem nagyok a különbségek. A diákok 65%-a tehát úgy érzi, hogy nagyjából a csoport átlagos szintjén van. Ez nagyon jó eredmény, tekintve, hogy az iskolai szakkör esetén ugyanez a mérőszám 30%, az informatika órai programozás esetén pedig 7%.



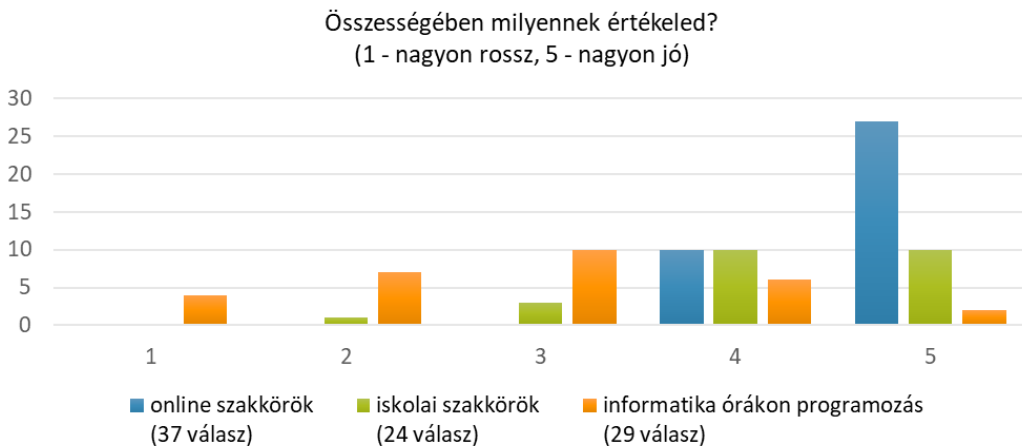
7. ábra: A társak szintje.

A szakmai és gyakorlati szempontok mellett fontosak a diákok közötti kapcsolatok, a közösségi élmény is. Az erre vonatkozó kérdésben az iskolai szakkörök kapták a legjobb értékelést (átlag: 4,54), és az informatika órák (átlag: 3,90) is kicsit jobb pontszámokat kaptak, mint az online szakkörök (átlag: 3,86). Tehát megerősítést nyert az is, hogy a diákoknak értékesebbek a közvetlen személyes kapcsolatok, mint a virtuálisak. Megjegyezzük, hogy szerintünk online is lehet nagyon jó közösséget építeni, és úgy gondoljuk, hogy az online szakköreink platformján még vannak kihasználatlan lehetőségek, amelyekre a jövőben több figyelmet fordítunk.



8. ábra: A közösség értékelése.

A teljesség kedvéért az általános értékelés eloszlását is ábrázoljuk az alábbiakban, bár ebből a fentiekhez képest új következtetést nem vonhatunk le (átlagok: 4,73; 4,21; 2,83).



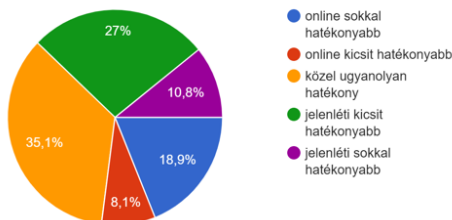
9. ábra: A közösség értékelése.

### 4.3. Az online szakkörök előnyei és hátrányai

Nagyon érdekes módon megosztott válaszokat kaptunk arra a kérdésre, hogy online vagy jelenléti foglalkozásokon lehet hatékonyabban tanulni (programozást). Kifejezetten fontos számunkra, hogy csupán 11% mondja azt, hogy jelenléttel sokkal hatékonyabban lehet tanulni, ezzel szemben 19% véli úgy, hogy online sokkal hatékonyabban lehet tanulni. Mivel 35% szerint közel ugyanolyan hatékony a két forma, ezért összesen 62% állítja azt, hogy az online oktatás legalább olyan hatékony, vagy még hatékonyabb, mint a jelenléti. A kényelem szempontjából elsöprő sikere van az online szakköröknek, 81% szerint kényelmesebb az online, míg csak 3% részesíti előnyben a jelenléti órákat. Az online szakkör előnyeire adott szabad szöveges válaszok egy jelentős része is erről szólt, például: „Nem megy el idő utazással”, „Könnyebben megoldható az időpont, nem megy el idő utazással, saját gépet tudunk használni”; „Sokkal kényelmesebb, hogy otthonról lehet tanulni”; „Nem kell utaznom, praktikus, gyors, néhol személytelenebb, megmarad az elhangzott anyag, lehet hét közben is”.

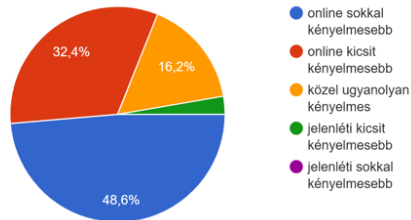
Az online szakkörökön mennyire lehet hatékonyan tanulni, jelenléti foglalkozásokhoz hasonlítva?

37 válasz



Az online szakkörökön mennyire lehet kényelmesen tanulni, jelenléti foglalkozásokhoz hasonlítva?

37 válasz



10. ábra: Az online tanulás hatékonysága és kényelme a jelenlétihez viszonyítva.

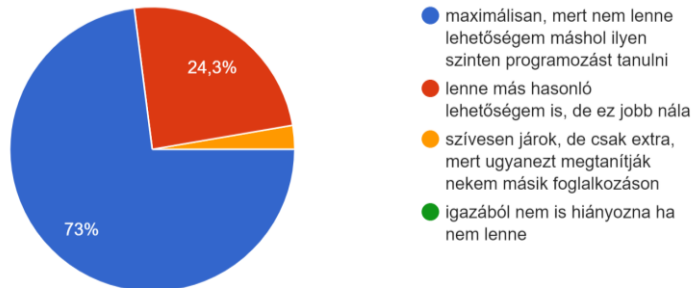
Az online szakkörök előnyeiről szóló szabad szöveges válaszokban sokan említették azt, hogy nincsenek földrajzi korlátok, és az egyszerűen készíthető felvétellel az időbeli akadályok is részben kezelhetők („Az ország bármely pontjáról be lehet kapcsolódni, ottbón kicsit kényelmesebb.”; „Részét tudnak venni azok is, akiknek utazással együtt nem lenne rá idejük, a szakkör közbeni helyváltoztatás kivitelezhető, a felvételek visszanezézhetőek.”; „Az, hogy nem kell elmenni a helyszínre, és fel is lehet egyszerűen venni”; „Több ember részt tud venni rajta, nincs az a probléma, hogy sokan nem ugyan ott élnek.”). Az otthoni megszokott környezet mellett még az időbeosztásról és a megspórolt utazási időről szól a válaszok jelentős része („Nem megy el idő utazással”; „Könnyebben megoldható az időpont, nem megy el idő utazással, saját gépet tudunk használni.”; „Nem megy el időm az utazással, így könnyebben bele tudom építeni a napirendembe.”; „Kényelmesebb, rugalmasabban lehet tartani az órát”; „Nem kell utazással eltölteni az időt, előbb hazá lehet menni, rugalmasabb a többi program szempontjából”).

A hátrányok között a legtöbb válasz az esetlegesen felmerülő technikai problémákat teszi szóvá („Előfordulhat, hogy egyeseknek technikai gondjaik vannak, ez engem nem érint.”; „Vannak olyanok, akiknél rossz az internet, vagy mikrofon és az eléggé zavaró tud lenni.”; „Kicsit személytelenebb, esetleg lehetnek technikai problémák”; „Néha nehezebb kezelni a felületeket, amiken dolgozunk, általában pillanatnyi technikai okok miatt”). Többen felvetik a személyes együttlét hiányát („Személy szerint én kicsit hatékonyabbnak érzem az oktatást, ha személyesen tudok kommunikálni a tanárokkal és a többi diákkal.”; „Nagyobb távolságok vannak az emberek között. Nem biztos, hogy lesz alkalom találkozni”; „Sokkal jobb, ha egy teremben ülök azokkal, akikkel együtt dolgozom”). Érdekes és fontos visszajelzés nekünk az, hogy számos válaszadó szerint online nehezebb a figyelem fenntartása („Online könnyebben elkalandozik az ember figyelve, nehezebb koncentrálni”; „Élőben könnyebb figyelni, megérteni az anyagot.”), bár van ellenvélemény is („Ha az ember fáradtabb, talán könnyebb elbambulni, és nem dolgozni. Bár még így is sokkal hatékonyabb a számomra az online óra, hiszen jelenlétinél nem csak akkor nem tudok figyelni, amikor fáradt vagyok, hanem úgy általában.”).

Közvetlen kérdésként is feltettük a kérdőívben az elsődleges hipotézisünket, miszerint az online szakkörök hiánypótlóak. Az erre érkezett válaszok magukért beszélnek.

## Hiánypótlónak érzed az online programozás szakköröket?

37 válasz



11. ábra: Hiánypótlóak-e az online szakkörök?

## 5. Konklúzió

A bemutatott eredmények alapján kijelenthetjük, hogy az online tehetséggondozó programozás szakkörök hiánypótlóak hazánkban, mert sok diáknak nem lenne más lehetősége a saját szintjének megfelelő csoportban programozást tanulni. Emellett a résztvevők a szakmai színvonalat kimondottan magasnak értékelik, és számos gyakorlati előnyét is megbecsülik. Néhány fejlesztendő problémára is fény derült a kutatás által: az online órákon a figyelem fenntartásához be kell vetnünk új eszközöket, és a közösségépítésbe a Discord platformon jóval több energiát kell fektetni, hogy megközelítsük a jelenléti foglalkozások közösségi élményét.

## Irodalom

1. A. R. Masonbrink, E. Hurley: *Advocating for children during the COVID-19 school closures*. In: *Pediatrics* 146.3 (2020)
2. Z. Ma, S. Idris, Y. Zhang, L. Zewen, A. Wali, Y. Ji, Z. Baloch: *The impact of COVID-19 pandemic outbreak on education and mental health of Chinese children aged 7–15 years: an online survey*. In: *BMC pediatrics* 21.1 (2021), pp. 1-8
3. X. Yang, M. Zhang, L. Kong, Q. Wang, J. C. Hong: *The effects of scientific self-efficacy and cognitive anxiety on science engagement with the question-observation-doing-explanation model during school disruption in COVID-19 pandemic*. In: *Journal of Science Education and Technology* 30.3 (2021), pp. 380-393
4. A. E. Clark, H. Nong, H. Zhu, R. Zhu, R: *Compensating for academic loss: Online learning and student performance during the COVID-19 pandemic*. In: *China Economic Review* 68 (2021), 101629
5. K. N. Kollár: *Az online oktatás tapasztalatai és gyakorlata a pedagógusok nézőpontjából*. *Iskolakultúra* 31.2 (2021), pp. 23-53
6. O. B. Adedoyin, E. Soykan: *Covid-19 pandemic and online learning: the challenges and opportunities*. In: *Interactive Learning Environments* (2020), pp. 1-13
7. Á. Erdősné Németh: *A Logo-tól az informatikai olimpiáig informatikai tehetséggondozás a középiskolában*. Doktori értekezés, Eötvös Loránd Tudományegyetem (2019)
8. *Alphacademy – A gondolkodás iskolája*. <https://alphacademy.hu/> (utoljára megtekintve: 2021.11.15.)
9. L. Nikházy: *A Problem-based Curriculum for Algorithmic Programming*. In: *Central-European Journal of New Technologies in Research, Education and Practice* (2020), pp. 76-96



10. *Discord*.  
<https://discord.com/> (utoljára megtekintve: 2021.11.15.)
11. L. Nikházy: *Az első ProgTábor: egy új tehetséggondozó program kezdete az algoritmikus programozásban*. In: L. Zsakó, P. Szlávi (szerk.) *INFODIDACT 2020*, Budapest, Webdidaktika Alapítvány, pp. 135-147  
<http://konferenciak.inf.elte.hu/infodidact/InfoDidact20/Infodidact2020.pdf> (utoljára megtekintve: 2021.11.15.)
12. *Nemes Tibamér Nemzetközi Informatikai Tanulmányi Verseny - Programozás kategória*.  
<http://nemes.inf.elte.hu/> (utoljára megtekintve: 2021.11.15.)
13. *Informatika Olimpiai Válogatóverseny*.  
[http://tehetseg.inf.elte.hu/valogatok/valogatok\\_main.html](http://tehetseg.inf.elte.hu/valogatok/valogatok_main.html) (utoljára megtekintve: 2021.11.15.)



# A középiskolai tanulmányok hozadéka a számítógépes gondolkodás perspektívájából

Osztían Pálma Rozália<sup>1</sup>, Kátai Zoltán<sup>2</sup>, Osztían Erika<sup>3</sup>

{<sup>1</sup>osztian.palma, <sup>2</sup>katai\_zoltan, <sup>3</sup>osztian}@ms.sapientia.ro  
SAPIENTIA EMTE

**Absztrakt.** A diákok problémamegoldó képességének és kreativitásának fejlődése szoros kapcsolatba hozható a számítógépes gondolkodás fogalmával. Ennek a képességnek a fontosságára 2006-ban Jeanette Wing hívta fel a figyelmet, aki szerint ez a negyedik alapképesség, amellyel minden embernek rendelkeznie kell napjainkban. Az sem titok, hogy ezt a képességet már egészen kiskorban fejleszteni kellene, annak érdekében, hogy elősegítsük az egyének számítógépes problémamegoldó képességének fejlődését is.

Jelen kutatásunkban azt kívántuk felmérni, hogy a jelenlegi oktatás, pontosabban szólva a középiskolai tanulmányok, a Romániában működő tanulmányi rendszer és ehhez kapcsolódó felmérések milyen mértékben járulnak hozzá a számítógépes gondolkodás mint alapképesség, és a diákok problémamegoldó képességének fejlesztéséhez. Arra voltunk kíváncsiak, hogy a tanulók érettségi átlaga, matematika, informatika és fizika tantárgyakból szerzett érettségi eredménye szoros kapcsolatban áll-e a számítógépes gondolkodás mint alapképesség felméréséhez kapcsolódó teszten elért pontszámaikkal.

**Kulcsszavak:** számítógépes gondolkodás, megértés, fejlődés, algoritmika, oktatás, problémamegoldás

## 1. Bevezető

Pedagógusként egyik célunk, hogy mindennapi oktatói munkánkkal hozzájáruljunk a diákok ismereteinek bővítéséhez. Sokszor azonban azt tapasztalhatjuk, és a diákok visszajelzéseiből azt észlelhetjük, hogy az iskolai tanterv nem feltétlen járul hozzá az életben előforduló problémák könnyedebb megoldásához, hiszen hiányoznak belőle azon kulcsfontosságú elemek, melyek a *problémamegoldó* készség fejlesztéséhez szükségesek. Fontos szem előtt tartanunk, hogy nem elég csupán tanítani, adatot közvetíteni, hanem azt is szem előtt kell tartani, hogy milyen információtartalmat hordoz magában az a bizonyos adat, illetve milyen megközelítésekkel, módszerekkel van bevezetve az iskolai tantervbe.

Kutatócsoportunk hosszú ideje foglalkozik az oktatás és ezen belül az algoritmusvizualizációk hatékonyságának vizsgálatával [1], [2], [3], [4] és a tanulók számítógépes gondolkodás képességének fejlesztésével. Eddigi munkáink során olyan oktatói anyagokat igyekeztünk kidolgozni, melyek a szokásostól eltérő módon próbálják a diákokkal megismertetni az informatika különböző területeit. Munkánk során kiemelt hangsúlyt fektettünk a *számítógépes gondolkodás* fontosságára és ennek fejlesztésére. Ennek fontosságára Pólya is felhívta a figyelmet [5], aki azt fogalmazta meg, hogy amennyiben a tanulók képesek a komplexebb, összetettebb feladatokat elemi szintre bontani (felépíteni a megoldást az egyszerűtől a bonyolult fele) a problémamegoldási képesség is jobban fejlődik. Erre egy Angliában kiadott tantervben is találhatunk utalásokat [1], amely kihangsúlyozza, hogy mielőtt komoly programozásra kerülne sor, fontos előbb a tanulók problémamegoldó képességét fejleszteni.

Kutatócsoportunk egy része már 2012-től alkalmaz olyan stratégiákat programozással kapcsolatos tanórákon a Sapientia Erdélyi Magyar Tudományegyetem Marosvásárhelyi karán, amelyek nem

csak a különféle programozási nyelvek, hanem a számítógépes gondolkodás, és ezen belül az algoritmikus képesség fejlesztésére is nagy hangsúlyt fektetnek. A tanulók számítógépes gondolkodásának felmérésére gyakran alkalmazunk ehhez kapcsolódó feladatsorokat, melyeket főként az elsőéves hallgatók körében szoktunk alkalmazni. A frissen felvételizett egyetemisták számítógépes gondolkodás teszten elért eredményei azonban azt sugallják, hogy sok esetben a középiskolákból érkező diákok nincsenek tisztában a számítógépes gondolkodás fogalmával és néha hiányzik vagy nagyon alacsony szintű ezen képességük. Minden tanévben kiemelt hangsúlyt fektetünk arra, hogy felmérjük az újonnan érkezett egyetemisták problémamegoldó képességét, a 2021/2022-es tanévben úgy döntöttünk, hogy megvizsgáljuk a hallgatók számítógépes gondolkodás képessége és az érettségien szerzett minősítése közti kapcsolatot.

A Sapientia Erdélyi Magyar Tudományegyetem Marosvásárhelyi karán 2021/2022 tanévet megelőzően minden reál szak (Automatika és alkalmazott informatika, Gépészmérnöki, Mechatronika, Informatika, Számítástechnika és Infokommunikációs rendszerek), felvételi jegyének két komponense közé az érettségi átlag (50%) és az érettségien matematikából/fizikából/informatikából írt legnagyobb jegy tartozott (50%). A 2021/2022-es tanévtől kezdődően a korábbiakhoz hasonlóan ez a két komponens alkotja felvételi jegyet, ezáltal azonban a következő eloszlásban: 25% érettségi átlag, 75% matematikából/fizikából/informatikából írt legnagyobb jegy.

Jelen kutatásunkban a számítógépes gondolkodás képesség szintjének mérését tűztük ki célul, mindezt a középiskolai tanulmányok eredményeit vizsgálva. Arra kerestük a választ, hogy milyen összefüggés fedezhető fel a diákok érettségi átlaga, matematika, valamint választható tantárgy (informatika vagy fizika) jegyei és számítógépes gondolkodása között. Továbbá, arra, hogy van-e különbség különböző előzetes programozási tapasztalattal rendelkező diákok számítógépes gondolkodás képesség szintje között.

## 2. Szakirodalmi áttekintő

A számítógépes gondolkodás eredete egészen Papert nevéhez húzódik vissza [6], ennek ellenére az áttörés és a „számítógépes gondolkodás mindenkinek” mozgalom Jeanette Wing [7] nevéhez fűződik, aki 2006-ban úgy vélekedik erről mint egy univerzális magatartás és képesség, amellyel nem csak az informatikusoknak, hanem mindenkinek rendelkeznie kell.

A digitális eszközök széleskörű elterjedésével az elmúlt évek során a számítógépes gondolkodás fejlesztése egyre nagyobb hangsúlyt kapott az oktatásban. Ezzel kapcsolatosan két irányzat is szárnyra kapott, amelyek közül az egyik az új tantárgyak bevezetését tűzte ki célul, míg a másik a meglévő tantárgyak célszerű kibővítését olyan elemekkel, amelyek elősegítik ezen képességek fejlesztését. Evans és kutatótársai [8] arra a következtetésre jutottak, hogy azok a tantervi programok, amelyek nem fókuszálnak expliciten a tanulók számítógépes gondolkodásának gyarapításával, nem járulnak hozzá ennek a képességnek a fejlesztéséhez. Mások azt fogalmazták meg, hogy bár lehetséges, hogy az iskolák nem fókuszálnak ennek fejlesztésére, mivel a számítógépes gondolkodás képességnek van egy multidiszciplináris vonatkozása is, a többi tantárgy járulékos hozadékaként is fejlődhet [9]. Sokan azt is kutatták, hogy miként tudnak azon diákok megbirkózni különböző típusú informatikai feladatokkal, akik iskolai tantervében nem szerepel programozás oktatás [10]. Lewandowski és munkatársaihoz hasonlóan [10] mi is minden évben fontosnak tartottuk a reál szakos (Automatika és alkalmazott informatika, Gépészmérnöki, Mechatronika, Informatika, Számítástechnika és Infokommunikációs rendszerek) elsőéves hallgatók számítógépes gondolkodás képességének felmérését. Ezáltal, minden tanév elején sikerült egy átfogó képet kapni az évfolyam számítógépes gondolkodásának szintjéről.

Továbbá, fontos megemlíteni, hogy a *Computer Science Teachers Association* (CSTA) által kiadott standardok [1] is utalnak arra, hogy az algoritmusokhoz és programozáshoz kapcsolódó tantervek

fontos részét képezik a K-12 oktatási formának. Azt tapasztalhatjuk, hogy a számítógépes gondolkodással kapcsolatos feladatok egészen kiskortól jelen kellene legyenek a tanulók tantervi összeállításában, és akár már elemi iskolában be lehet vezetni az ehhez kapcsolódó problémamegoldással kapcsolatos feladatokat. Ezzel szemben, Romániában az oktatás jelenlegi formája nem járul hozzá feltétlen a diákok számítógépes gondolkodásának fejlődéséhez [2], holott ez a fogalom mindenki számára ismeretes kellene legyen.

A Sapientia Erdélyi Magyar Tudományegyetemen folyamatosan arra törekszünk, hogy a számítógépes gondolkodás hiányát némiképp próbáljuk pótolni, illetve kiemelt hangsúlyt fektetünk arra, hogy a hallgatók problémamegoldó képességét is fejlesszük a tudás átadása mellett. Továbbá, jelenlegi kutatásunk azt a célt szolgálja, hogy közelebbről is szemügyre vegyünk milyen összefüggés van a középiskolai tanterv, a középiskolai megmértetések, a romániai érettségi és a tanulók számítógépes gondolkodás képességének szintje között.

### 3. Kutatási kérdések

Az előzetes szakirodalmi kutatások alapján azt feltételeztük, hogy a diákok érettségi minősítései korrelálnak a számítógépes gondolkodás teszt eredményeivel, lineáris kapcsolatot alkotva. Ennek kapcsán a következő hipotéziseket fogalmaztuk meg:

- **H<sub>1</sub>:** Összefüggés van a diákok érettségi minősítése és a számítógépes gondolkodás teszten elért pontszámok között
- **H<sub>2</sub>:** A diákok programozási tapasztalata hatással van a számítógépes gondolkodás teszten elért eredményekre
- **H<sub>3</sub>:** Összefüggés van a feladat típusa (komplexitása) és az előzetes programozási tapasztalat között

### 4. Módszertan

A kutatás egy számítógépes gondolkodás tesztet, valamint a hallgatók érettségi minősítésének elemzését foglalja magába. A kísérlet megvalósítására a 2021/2022-es tanévben került sor, a Sapientia Erdélyi Magyar Tudományegyetem Marosvásárhelyi karán.

#### 4.1. Résztvevők

A kísérleten összesen 137 hallgató vett részt, melyek közül 26 nem töltötte ki megfelelően az érettségien szerzett minősítéssel kapcsolatos kérdőívet, így az ezzel kapcsolatos kutatási kérdések során 111 hallgató eredményeit összesítettük. A résztvevők a Sapientia Erdélyi Magyar Tudományegyetem elsőéves hallgatóiból tevődtek össze, 5 különböző reál szakról: Automatika, Gépészmérnöki, Informatika, Mechatronika és Számítástechnika.

A résztvevőket az előteszt eredmények alapján három kategóriába soroltuk az előzetes programozási tapasztalatok alapján: *Nincs előzetes programozási tapasztalat* (NP: egy évet sem tanultak programozást a középiskolás évek során), *Alap programozási tapasztalattal* (BP: 1, 2 vagy 3 évet tanultak programozást a középiskolás évek során, természettudomány osztály diákjai; heti 1-2 programozás óra) és *Magas programozási tapasztalattal* (HP: 4 évet tanultak programozást a középiskolás évek során, matematika-informatika osztály diákjai; heti 5-7 programozás óra) rendelkező diákok. Az eredményeket a fent említett programozási tapasztalatok függvényében is elemeztük.

#### 4.2. Kutatási eszközök

Az előzetes kérdőív és a számítógépes gondolkodás teszt a *Quizizz* online platform segítségével valósult meg. Az előzetes kérdőív összesen 5 kérdésből állt: 1 személyes adatok feldolgozására

vonatkozó kérdés, 3 demográfiai adatokra vonatkozó kérdés és 1 előzetes programozási tapasztalattal kapcsolatos kérdés (Hány évet tanultál programozást középiskolában?).

A résztvevők számítógépes gondolkodásának felmérését célzó teszt 10 kérdést tartalmazott, amelyeket programozási tapasztalattól függetlenül meg tudtak a hallgatók válaszolni. A tesztkérdések háttérül szolgáló feladatokat a *Bebras* verseny weblapjáról [2] válogattuk ki, egészen pontosan ennek magyar megvalósulásáról, az *e-hód* [4] online platformról (7).

### 4.3. Bebras feladatokra épülő számítógépes gondolkodás teszt

Ahogy azt Dagiene és kutatótársai említik egy 2016-os cikkükben [11], a Bebras oldalon [3] található feladatok kiváló lehetőséget biztosítanak olyan való életből vett példák modellezésére, amelyeket már egészen kiskortól képes megérteni egy diák. A Bebras közössége kijelentette, hogy fontos a számítógépes gondolkodást már kiskortól fejleszteni. Ezt próbálják elősegíteni az oldalon megjelenő feladatok, melyek jellegükben is igen változatosak. Megtalálhatóak többek között olyan feladatok a Bebras kollekcióban, amelyek való életből vett problémákat szemléltetnek játékosan, amelyeket számítógép segítségével lehet megoldani, de sok más logikai és elemzéssel kapcsolatos feladat is jelen van.

Az oldalon öt különböző kategória szerint választhatunk feladatokat: absztrakció (*abstraction*), algoritmikus gondolkodás (*algorithmic thinking*), felbontás (*decomposition*), kiértékelés (*evaluation*), általánosítás (*generalization*). Az évek során számos alkalommal döntöttünk úgy, hogy az elsőéves hallgatók számítógépes gondolkodás képességének felmérése érdekében a feladatokat a Bebras oldaláról válogatunk. Ezeket többnyire változó kategóriájú feladatok közül választottuk ki, de minden alkalommal kiemelt hangsúlyt kaptak az algoritmikus gondolkodással kapcsolatos feladatok, a kiértékelés és az általánosítás.

Mivel a Bebras oldalon a feladatok angol nyelven elérhetők és egyetemünkön magyar nyelvű oktatás folyik, a feladatok fordításához az *e-hód* online oldalt [4] használtuk segítségként. A kiválasztott feladatokat minden alkalommal átformáltuk, esetenként pedig további kérdésekkel bővítettük ki.

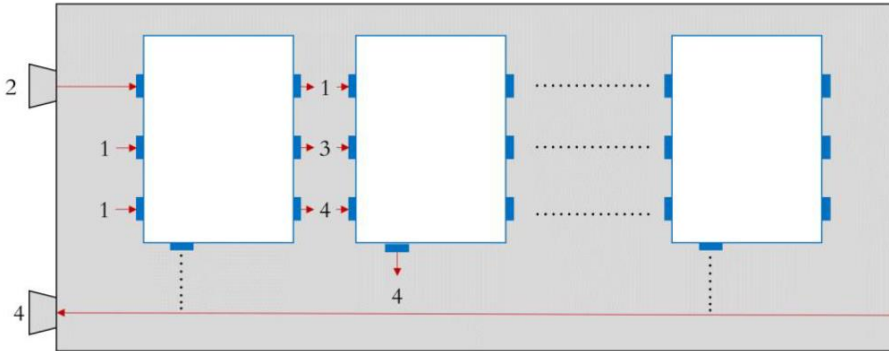
A feladatok sokszínűségének és jellegének köszönhetően mi is egyetértünk Dagiene és kutatótársai megfogalmazásával [11], miszerint a Bebras feladatok és a számítógépes gondolkodás képesség fejlesztése kéz a kézben jó eredményekhez vezethet, amennyiben alkalmazva van a tantervben. Hasonló következtetéshez jutott Hubwieser és munkatársai [12] illetve Csapó [13] is, akik szerint a Bebras feladatok segítségével érdekesen és vonzón lehet bemutatni a számítástechnika különféle területeit.

A Bebras és *e-hód* feladatoknak hatékonyságát, a diákok gondolkodására való ráhatás céljával Pluhár és Gellér is hangsúlyozza [14]. Továbbá, Pluhár és munkatársai kiemelik ezen feladatok szerepét az algoritmikus gondolkodás fejlesztésében, az informatika iránt való érdeklődés felkeltésében, az informatikával kapcsolatos félelmek, negatív érzések feloldásában és az informatika területei sokszínűségének, felhasználási lehetőségeinek kifejezésében [15].

Éppen ezért, a 2021/2022-es tanévben, amikor előkészítettük az előzetes felméréshez szükséges tesztet, előző évekhez hasonlóan a Bebras feladatai mellett döntöttünk. Ezúttal három különféle feladatot választottunk ki:

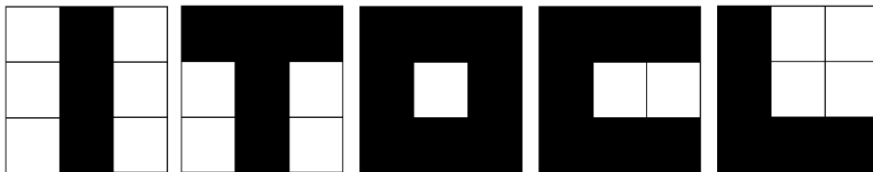
- **Számológép feladat:** A teszt első feladata egy számológéphez hasonlóan működő problémát szemléltetett (1. ábra: Számológép feladat: bemeneti és kimeneti egységek). A számológépnek három bemeneti egysége volt (a, b és c), melyek értékeinek megadását követően a számológépnek követnie kellett a kifizűzött algoritmus lépéseit, majd ennek kiértékelésekor visszaszolgáltatta a végső eredményt. A feladathoz 4 kérdést társítottunk (0 melléklet), két meghatározott értékhez (bemenet:  $a=4$ , majd  $a=7$ ) tartozó kiértékelés eredményét, egy műveletsort, valamint egy általánosításról szóló kérdést (bemenet:  $a=n$ ). Ennek a feladatnak a megoldásához a hallgatóknak egy jól meghatározott algoritmust kellett követniük lépésről-lépésre ahhoz, hogy megadják a

végző eredményt. Az utolsó előtti alpont (műveletsor) a kiértékelés mellett felbontásokat is tartalmazott, hiszen a műveletsor részeredményeit egy-egy részfeladatként lehet tekinteni. A feladat utolsó alpontja, az általánosítás, már az algoritmusban való elmélyülést vizsgálja, vagyis azt, hogy a résztvevő milyen mértékben tud azonosulni a feladattal, rájönni a feladat logikájára.



1. ábra: Számológép feladat: bemeneti és kimeneti egységek

- Hőtérkép feladat:** A második feladatban fontos szerepet kap a kétdimenziós reprezentáció, amely alapján különböző betűk különbségkártyáit kell meghatározni (2. és 3. ábra). A különbségkártya az adott kép minden képpontjához egy értéket rendel. Az érték azt mutatja meg, hogy a többi kép esetében hánynál szerepel az adott helyen ugyanaz a képpont. Ehhez a feladathoz három kérdést társítottunk (0 melléklet), melyek közül az első két kérdésben egy-egy konkrét betű különbségkártyájára kérdezzük rá, míg a harmadik kérdés esetén a hibás különbségkártyát kell meghatározni (azt a kártyát, amelyik nem tartozik egyik betűhöz sem).



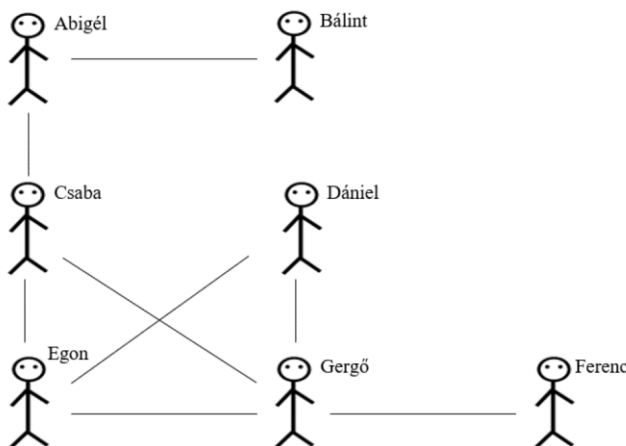
2. ábra: Hőtérkép feladat: betű reprezentáció

0	3	1
1	1	3
1	4	1

3. ábra: Hőtérkép feladat: I betű különbségkártyája

- Népszerűség feladat:** Az utolsó feladat egy újabb mindennapi probléma feltárását próbálja szemléltetni, mely az egymás közti kommunikációra, a szociális hálókra épül. Az általunk kiválasztott Népszerűség (Popularity) feladat (0 melléklet) Dagiene cikkjében [11] is kiemelkedő szerepet kapott, hisz véleménye szerint is a komplexebb, hálózatokkal, gráfokkal kapcsolatos problémák szemléltetésének egyik legjobb módja, ha a szociális hálók, barátságok segítségével közelítjük meg a feladatot. A feladat hat barát egymás oldalára írt bejegyzéseire épül. Minden személy a barátok közül egy-egy mottóval (bejegyzéssel) mutatkozott be, melyet megosztott a

barátaival a hálózaton. Így a barátaik oldalán is megjelent a mottó. Minden személy a saját oldalán és a barátaik oldalán lévő mottókat láthatta. A kérdések ezúttal arra vonatkoznak, hogy kik látják egy adott diák bejegyzését (1), hány olyan diák van, akinek a bejegyzését pontosan 4 diák látja (2), illetve utolsó kérdésként abban voltunk érdekeltek, hogy ilyen kapcsolatokat kell a hálón felbontani, ahhoz, hogy egy bizonyos bejegyzés ne jusson el egy másik diákhoz (3).



4. ábra: Népszerűség: baráti kapcsolatok hálózata

#### 4.4. A kutatás menete

A számítógépes gondolkodás teszt (továbbiakban SZG teszt) és az előzetes kérdőív kitöltése 40 percet vett igénybe, melyet minden diák ugyanazon időben töltött ki az egyetem erre kijelölt előadótermeiben. Ezt követően egy újabb kérdőív keretén belül megkérdeztük a hallgatókat az érettségien szerzett minősítésükről, ahol az érettségi átlagra, a matematika, valamint a választott tantárgyból (fizika/informatika) szerzett minősítésekre voltunk kíváncsiak.

Ezt követően összesítettük a hallgatók teszten elért eredményét, érettségi minősítését és kezdetét vette az elemzés.

### 5. Eredmények

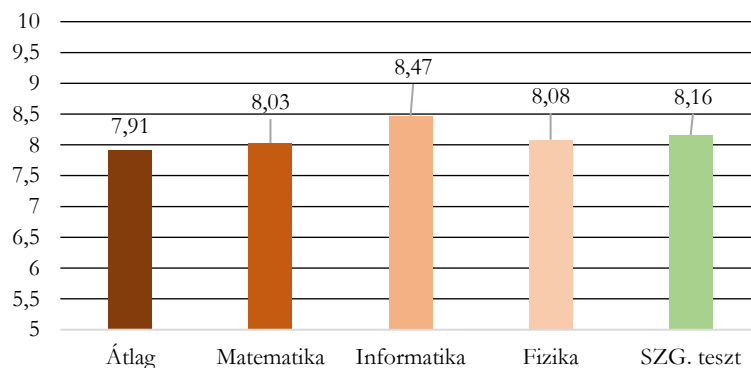
Az eredmények elemzésekor elsősorban az érettségi minősítésekre, a SZG teszten elért eredményekre, az előzetes programozási tapasztalat mértékére, valamint az imént felsorolt három komponens közti korrelációra fektettünk hangsúlyt. A vizsgálat során a komponensek közti szoros és gyenge Pearson féle korrelációt kívántuk felmérni és páronkénti t-próbákat végeztünk.

Elsőként, kíváncsiak voltunk az említett két komponens – érettségi eredmény és SZG teszt – átlagaira. Az érettségi esetén 4 jegyben voltunk érdekeltek: érettségi átlag, matematika és szakspecifikus (informatika vagy fizika) jegy.

Az 5. ábra mutatja, az érettségi átlagok közti eltérés nem számottevő ( $p=0.1488 > 0.05$ ), mely arra enged következtetni, hogy az érettségi tantárgyakon szerzett jegyek összességében kiegyensúlyozottak (átlag: 7.91, matematika: 8.03, informatika: 8.47, fizika: 8.08, SZG teszt: 8.16), és ezekkel összhangban lehet a SZG teszt nehézségi szintje is. Mivel az átlagok közül az informatika jegyek átlaga a legnagyobb azt feltételeztük, hogy az informatikából érettségiző diákok számítógépes gondolkodás teszt eredménye lesz magasabb, melynek oka az lehet, hogy az informatika oktatás jobban hozzájárul ezen képesség fejlesztéséhez.



## 5.1. Az érettségin és a SZG teszten elért eredmények összehasonlítása



5. ábra: Érettségi és SZG teszt eredmények

Az átlagok összehasonlítását követően kíváncsiak voltunk arra, hogy milyen mértékben korrelálnak a különböző érettségi tantárgyakon szerzett jegyek a SZG teszten elért eredményekkel. Ahhoz, hogy ezt megvizsgáljuk külön vizsgáltuk a korrelációt minden tantárgy esetén, mely során az alábbi táblázatban szereplő értékekhez jutottunk (1. táblázat).

Átlag	Algoritmika	0.25
Matematika		0.20
Informatika		0.20
Fizika		0.15

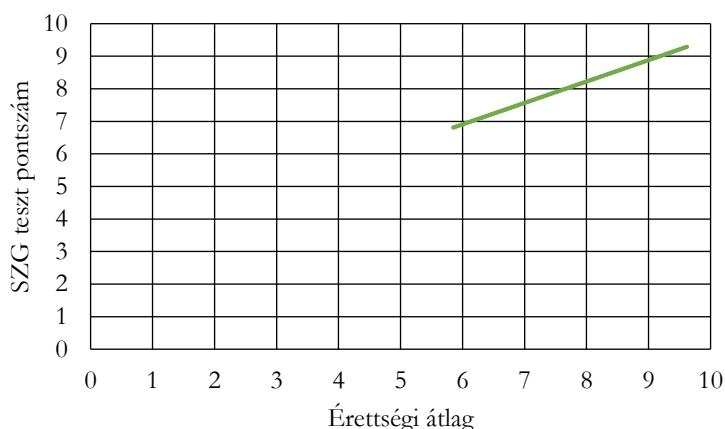
1. táblázat: Érettségi jegyek és SZG teszt közti korreláció

Amint az a táblázatban szereplő értékek alapján is észlelhető minden tantárgy és az érettségi átlag esetén is pozitív korreláció értékről beszélhetünk, ami arra utal, hogy a két változó közötti kapcsolat egyenes arányú és emelkedő. Azt azonban fontos megemlíteni, hogy a kiszámolt korrelációs értékek alacsonyak, amely arra utal, hogy az érettségi eredmények és számítógépes gondolkodás teszten mért teljesítmények kapcsolata gyenge (a 0-hoz közelít).

Mindez arra engedett következtetni, hogy bár magasabb érettségi eredményekkel minden bizonynyal jobb számítógépes gondolkodás teszt teljesítmény tartozna, a jelenlegi érettségi tanterv nem járul hozzá eléggé a diákok számítógépes gondolkodásának fejlesztéséhez.

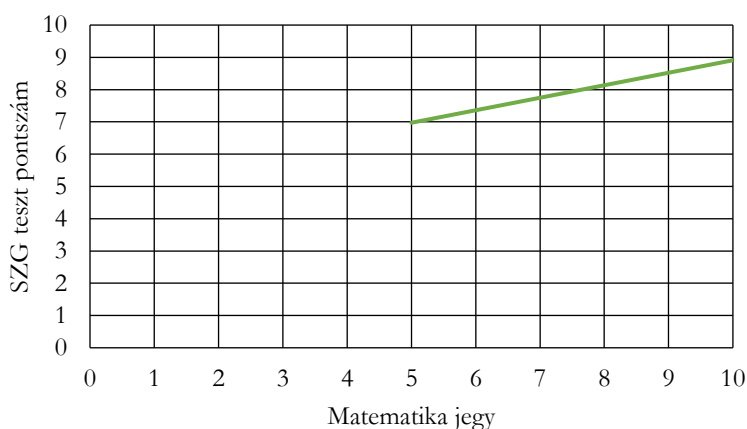
A továbbiakban kíváncsiak voltunk a különféle tantárgyakra vonatkozó korrelációs ábrákra. A legmagasabb korrelációs érték ( $R=0.252$ ,  $p=0.007 < 0.05$ ) az érettségi átlagok (6. ábra) során volt mérhető. Ezt a jelenséget annak tulajdonítottuk, hogy a számítógépes gondolkodás nem feltétlen csak az informatika, matematika, fizika vagy bármely más reál tudományokat tanító tantárgy keretén belül fejleszhető, hanem multidiszciplináris jellege miatt hatással van rá minden más tantárgy is, melyek segítenek a diákok problémamegoldó képességének fejlesztésében. Következtetésképp kijelenthetjük, hogy annak ellenére, hogy az egyetem felvételi jegy összetételét illetően az a javaslat érkezett, hogy az érettségi átlag kevesebbet számítson, mint a szakjegy, jelen esetben azt tapasztalhatjuk, hogy ez az érték jobban korrelál az SZG teszten elért eredményekkel. Mindez, annak is tudható,

hogy a szövegértés is, az aritmetika is fontos, multidiszciplinárisan fejleszhető képesség, ennek következtében ez az eredmény utalhat arra is, hogy nem indokolt, hogy a felvételi jegy összetételében szereplő komponensek arányának megváltoztatása egy jó váltás.



6. ábra: Érettségi átlag és SZG teszt eredmény közti korreláció

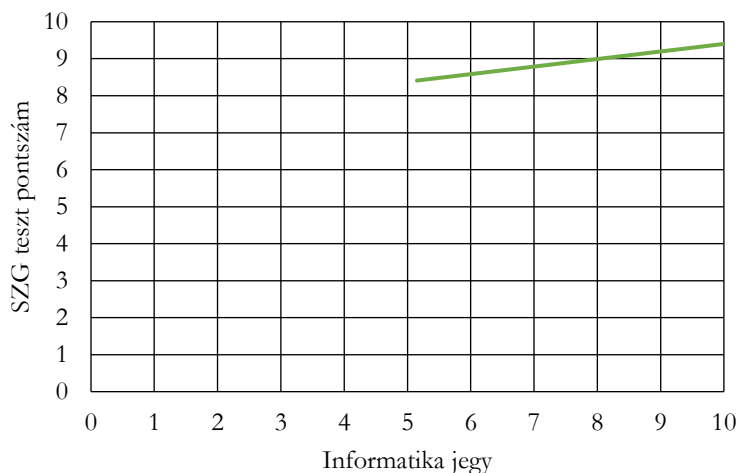
A matematika jegyek számítógépes gondolkodás képességre gyakorolt hozadékára különösen kíváncsiak voltunk. Azt feltételeztük, hogy a matematika nagymértékben befolyásolhatja és hatással lehet a tanulók problémamegoldó képességeire, hiszen bármilyen szakspecifikus osztályról vagy iskoláról legyen szó, mindenki tanul matematikát. Ennek ellenére az figyelhető meg, hogy nem különösebben járul hozzá ez a jobb számítógépes gondolkodás eredményekhez ( $R=0.202$ ,  $p=0.056 > 0.05$ ), és a jegyek eloszlását is a változatosság jellemzi (7. ábra). Minden bizonnyal a jelenlegi matematika oktatás sokkal jobban kellene összpontosítson arra, hogy a tananyag átadása mellett a diákok problémamegoldó képességet, kreativitását, logikus gondolkodását is fejlessze.



7. ábra: Matematika jegy és SZG teszt eredmény közti korreláció

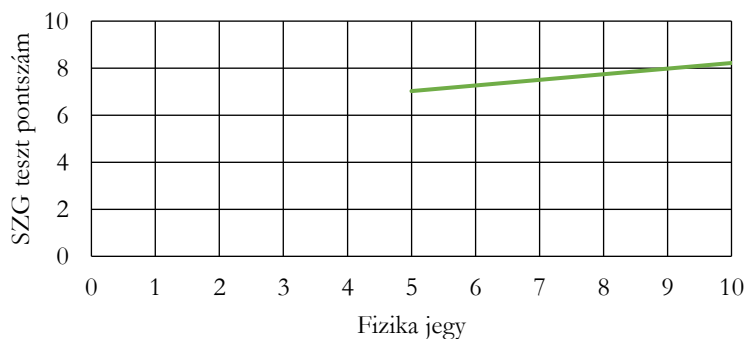
Hasonló jelenség figyelhető meg az informatika jegyek esetében is (8. ábra), azonban itt az SZG teszt eredmények magasabbak, mint más tantárgyak és az átlag esetén, sőt egy enyhe torlódás is érzékelhető a 9–10 pontszámok és minősítések esetén. Ennek ellenére a korrelációs együttható

értéke ( $R=0.209$ ,  $p=0.117>0.05$ ) nem jelentősen jobb, mint a matematika tantárgy esetén. Bár arra számítottunk, hogy az informatika jegy jobban fog korrelálni a SZG teszten elért pontszámmal, ez a feltételezésünk nem igazolódott be. Ennek oka az lehet, hogy az informatika érettségien megjelenő feladatok valószínűleg nem élezi ki a hangsúlyt a számítógépes gondolkodás képesség mérésre. Például a tétel nagyrésze tartalmaz programozás közeli feladatokat, amelyek nem feltétlen mérik elég hitelesen a diákok problémamegoldó és számítógépes gondolkodás képességét.



**8. ábra:** Informatika jegy és SZG teszt közti korreláció

A fizika tantárgy jegyei tekintetében (**9. ábra**) az előző értékekhez képest kisebb korrelációs együtthatóhoz jutottunk ( $R=0.151$ ,  $p=0.512>0.05$ ). Ennek következtében nem igazolódott be előzetes feltételezésünk, miszerint arra számítottunk, hogy a fizika oktatás gyakorlatias jellege miatt jelentősen hozzájárul a diákok problémamegoldó képességének fejlődéséhez is. Ehhez hasonlóan az informatika jegyek tekintetében is azt feltételeztük, hogy természeténél fogva jobban fog korrelálni a SZG teszt eredményeivel. E feltételezés ellenére az informatika minősítésekkel való korreláció sem kiemelkedően magas, sőt, az érettségi átlag amint azt korábban említettük felülmúlja.



**9. ábra:** Fizika jegy és SZG teszt közti korreláció

Mivel ezek a korrelációk pozitívak, de értékük messze túl alacsony ahhoz, hogy mindez számottevő legyen arra következtettünk, hogy a jelenlegi oktatási rendszer, ahogy az érettségi eredmények-

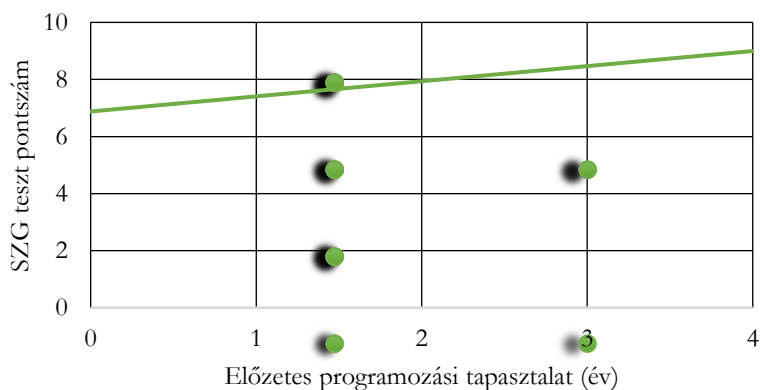
ben mérhető, nem járul hozzá elég szignifikánsan a számítógépes gondolkodás fejlesztéséhez, legalábbis nem tükröződik ezekben a minősítésekben.

Ez az eredmény összecseng Pluhár és munkatársai következtetésével [15], akik az ELTE Informatikai Karának Programming kurzusára járó hallgatók algoritmikus gondolkodását vizsgálták. E kutatók is arról számolnak be, hogy a vizsgált változók és az algoritmikus gondolkodás teszt pontszámai között nem találtak összefüggést.

Jelen kutatásban az érettségi jegyek minősítéseit mértük össze, azonban az is lehet egy magyarázat a gyenge korrelációra, hogy ezek a tételek, és úgy összességében az érettségi tematika nem tükrözi a számítógépes gondolkodás képesség szintjét, vagyis nem cseng össze kellőképpen az SZG teszttel.

## 5.2. Teljesítmény vizsgálata az előzetes programozási tapasztalat függvényében

A továbbiakban arra is kíváncsiak voltunk, hogy milyen mértékben befolyásolhatja a számítógépes gondolkodás teszten elért eredményeket a résztvevők előzetes programozási tapasztalata, vagyis azt kutattuk, hogy hogyan teljesítenek azon diákok, akik középiskolában 0 évet (NP), 1–3 évet (BP), illetve 4 évet (HP) tanultak programozást. Azt feltételeztük, hogy a több év programozás tapasztalat a diákok előnyére válik. Ez a feltételezésünk némiképp beigazolódott hiszen a programozási tapasztalat és az SZG teszt eredmény közti korreláció esetén az együttható  $R=0.421$  ( $p = 0 < 0.005$ ). Mindez nem utal szoros kapcsolatra a SZG teszt eredményei és a programozási tapasztalat között, azonban magasabb, mint az informatika érettségi eredményei kapcsán mért korreláció. Ez tulajdonképpen arra enged következtetni, hogy bár több év programozási tapasztalattal rendelkező diákok esetén jobb SZG teszt eredmény várható, ha az érettségien elért eredményeket tekintjük ez a jelenség nem igazolódik be. Ez ugyancsak ahhoz a tényhez vezethető vissza, hogy a négy év programozás oktatás során az iskolai tanterv csupán járulékos hozadéka az, hogy a diákok problémamegoldó képessége is nő. Ahhoz, hogy még jobb eredményeket érjünk el, szükséges az informatikaoktatás során is ezt elősegítő feladatokat és stratégiákat bevezetni.



10. ábra: Előzetes programozási tapasztalat és SZG teszt eredmény közti korreláció

Egy másik érdekes észrevételhez az előzetes programozási tapasztalat és az SZG tesztek összehasonlításakor jutottunk (10. ábra). Torlódási pontokat figyelhetünk meg úgy a 0 év programozási tapasztalattal, mint a 4 év programozási tapasztalattal rendelkezők esetén. Ez azzal magyarázható, hogy bár a több programozási tapasztalattal rendelkezők összességében jobb eredményeket érnek el,

vannak kivételek, akik annak ellenére, hogy nem tanultak informatikát, számítógépes gondolkodásuk tekintetében fel tudják venni a versenyt.

### 5.3. A feladat típusa és az előzetes programozási tapasztalat közti összefüggés

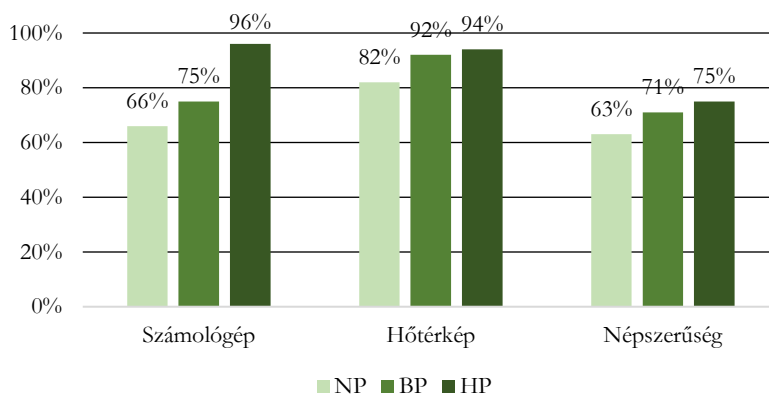
Végül, de nem utolsó sorban megvizsgáltuk az eredményeket a feladatok típusának függvényében kezdők (NP), középhaladók (BP) és haladók (HP) esetén. Az elemzést követően a különböző programozási tapasztalattal rendelkezők esetén a következő átlagokat kaptuk (2. táblázat):

Programozási tapasztalat	Átlag	Hallgatók száma
NP	69,75%	40
BP	78,82%	34
HP	89,36%	63

2. táblázat: Eredmények programozási tapasztalat szerint

Páronkénti t-próbák segítségével azt tapasztaltuk, hogy bár majdnem ugyanakkora eltérés észlelhető a kezdők és középhaladók közt, mint a középhaladók és haladók közt, az első kettő között nincs ( $p=0.0613$ ,  $p>0.05$ ), míg a második csoportpár között van szignifikáns különbség ( $p=0.0023$ ,  $p<0.05$ ). Kíváncsiak voltunk, hogy mi okozza a két csoport közti szignifikáns eltérést.

A 11. ábra mutatja, hogy a három feladat esetén kettőnél – Hőtérkép és Népszerűség – egy kiegyensúlyozott növekedés figyelhető meg a különböző előzetes programozási tapasztalattal rendelkezők eredményei között (82%-92%-94% és 63%-71%-75%). A Számológépes feladat esetén azonban a BP csoport jelentősen lemarad a HP csoport tagjaitól.

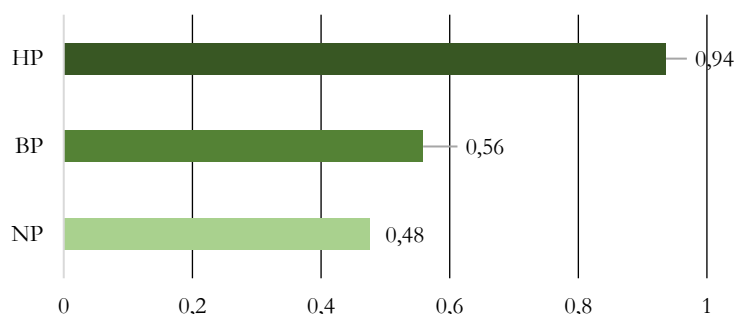


11. ábra: Teljesítmény feladattípusok és előzetes programozási tapasztalat szerint

Ahhoz, hogy magyarázatot kapjunk az eredmények alakulására alaposabban megvizsgáltuk a Számológép feladat alpontjait. Fontos tudni, hogy a feladatok közül ez volt az, amelyik általánosítás-sal kapcsolatos kérdést is tartalmazott: *Általánosítás: mit ad vissza a számológép, ha N-et adunk bemenetként?*

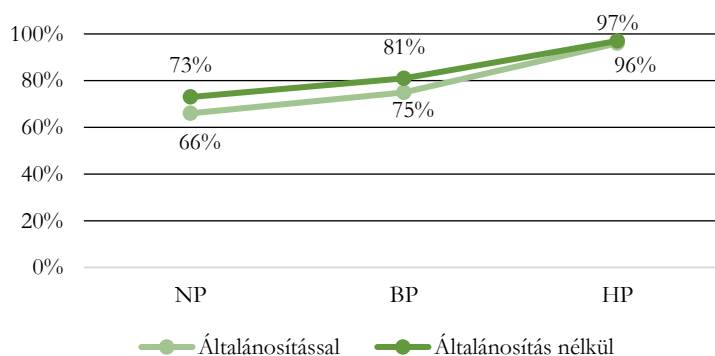
Erre a kérdésre adott válaszok pontszámát a 12. ábra mutatja. Megfigyelhető, hogy azon diákok, akik 4 év programozási tapasztalattal rendelkeztek sokkal nagyobb számban válaszoltak helyesen a feladat általánosítására vonatkozó kérdésre. Mindez, teljes mértékben tulajdonítható az előzetes 4 év középiskolai tapasztalatoknak (matematika-informatika osztály diákjai; heti 5-7 programozás óra), hiszen a feladat egy adott bemenet alapján, egy jól meghatározott algoritmust felhasználva kellett

visszaadjon egy értéket. Az algoritmus lépésről-lépésről való értelmezése és az eredmény meghatározása haladók esetén egy jelentősen egyszerűbb feladatnak számíthatott.



12. ábra: Számológép feladat – általánosítás

Az is belátható, hogy amennyiben az eredmények feldolgozásakor figyelmen kívül hagyjuk a Számológép feladat általánosítására vonatkozó válaszokat a csoportok közti pontszámok a következőképpen alakulnak (13. ábra).



13. ábra: Számológép feladat – általánosítással és anélkül

Míg az NP és BP csoportok esetén egy magasabb növekedést észlelhetünk (7% és 6%), mindez a HP csoport esetén csak egy 1%-os növekedésben mutatkozik meg. Ennek ellenére az eredményekre vonatkozó szignifikáns különbség nem tűnik el.

Ezek az eredmények arra engednek következtetni, hogy bár korábbi részeredmények arra utaltak, hogy az informatika oktatás nem járul hozzá elegendően a diákok számítógépes gondolkodás képességének fejlesztéséhez, a komplexebb, algoritmusokra épülő, általánosításokra vonatkozó feladatok esetén azon diákok, akik rendelkeznek némi programozási tapasztalattal jobban teljesítenek, mint azok, akik nem tanultak, vagy keveset tanultak programozást középiskolában. Annak érdekében, hogy ezt az ellentétet kiegyensúlyozzuk minden bizonnyal a többiek tantervi programjába is olyan feladatokat kellene bevezetni, amelyek javítják és összpontosítanak a tanulók számítógépes gondolkodás képességének fejlesztésére is.

## 6. Következtetések

Kutatásunkkal arra szerettünk volna választ kapni, hogy az elsőéves egyetemista hallgatók számítógépes gondolkodása milyen mértékben van összefüggésben a középiskolai tanulmányokkal és ezen belül az érettségi eredményekkel. Eredményeink arra engednek következtetni, hogy a középiskolában tanult tantárgyak járulékos hozadékként is fejlődhet ez a képesség, azonban az oktatásban ennél több potenciál is rejtőzik.

Annak érdekében, hogy az oktatás és a számítógépes gondolkodás közti korreláció magasabb legyen elengedhetetlen olyan eszközök és olyan feladatok bevezetése az oktatási folyamatba, amelyek elsősorban a diákok problémamegoldó képességére összpontosítanak és arra, hogy a bonyolultabb, komplexebb feladatokat egyszerű elemeire bontva oldják meg lépésről lépésre.

A romániai oktatási forma számos ismeretet próbál bemutatni és megtanítani a mai diákok számára, melyek nagymértékben hozzájárulnak az egyének konstruktív fejlődéséhez. Ezek azonban sokszor csak a tárgyilagos tudást bővítik és nem raktározódnak el a hosszútávú memóriában. A számítógépes gondolkodás készséget fejlesztő feladatok minden diák számára hasznosak lennének, és ezek oktatási tananyagba való bevezetése már egészen kiskortól fontos szerepet kellene kapjon.

Kutatócsoportunk elsődleges célja továbbra is, hogy az egyetemre érkező hallgatók tudását nem csupán az informatika szakterületét illetően bővítse, hanem olyan oktatási megközelítéseket alkalmazzon, olyan feladatokkal fűszerezve, amelyek az egyének számítógépes gondolkodását és ezáltal kreativitását, problémamegoldó képességét is fejlesztik. Az informatika szakos hallgatók mellett arra is hangsúlyt fektetünk, hogy más szakterületek hallgatói, illetve más korosztályú diákok számítógépes gondolkodását is próbáljuk fejleszteni. Meggyőződésünk, hogy ez a képesség hosszútávon az ő javukra is válhat, és pozitív eredményekhez vezethet, hiszen ez a tudás minden XXI. századi ember alapvető képessége [7].

## Irodalom

1. Kátai, Z., Osztian, E. (2021): *Improving AlgoRhythmic Teaching-Learning Environment by Asking Questions*. In: International Journal of Instruction, 14(2), 27-44
2. Kátai, Z. (2021): *AlgoRhythmic. Technologically and artistically enhanced computer science education*.
3. Nagy, E. J., Osztian, P. R., Cosma, C., Kátai, Z., Osztian, E. (2019, June): *Looking for the Optimal Interactivity Level in the AlgoRhythmic Learning Environment*. In: EdMedia+ Innovate Learning (pp. 106-114). Association for the Advancement of Computing in Education (AACE)
4. Osztian, P. R., Kátai, Z., Osztian, E. (2020, October): *Algorithm Visualization Environments: Degree of interactivity as an influence on student-learning*. In: 2020 IEEE Frontiers in Education Conference (FIE) (pp. 1-8). IEEE
5. Polya, G. (1954): *Mathematics and Plausible Reasoning*. Princeton, New Jersey: Princeton University Press
6. Papert, S. (1996): *An exploration in the space of mathematics educations*. In: Int. J. Comput. Math. Learn., 1(1), 95-123
7. Wing, J. M. (2006): *Computational thinking*. In: Communications of the ACM, 49(3), 33-35.
8. Evans, G. E., Simkin, M. G. (1989): *What best predicts computer proficiency?* In: Communications of the ACM, 32(11), 1322-1327
9. Feaster, Y., Ali, F., Zhai, J.; Hallstrom, J. O. (2014, June): *Serious toys: three years of teaching computer science concepts in K-12 classrooms*. In: Proceedings of the 2014 conference on Innovation; technology in computer science education (pp. 69-74). ACM

10. Lewandowski, G., Bouvier, D. J., Chen, T. Y., McCartney, R., Sanders, K., Simon, B.; Van-DeGrift, T. (2010): *Commonsense understanding of concurrency: computing students and concert tickets*. In: Communications of the ACM, 53(7), 60-70
11. Dagiéné, V., Sentance, S. (2016, October): *It's computational thinking! Bebras tasks in the curriculum*. In: International conference on informatics in schools: Situation, evolution, and perspectives (pp. 28-39). Springer, Cham.
12. Hubwieser, P., Mühling, A. (2015, April): *Investigating the psychometric structure of Bebras contest: towards measuring computational thinking skills*. In: 2015 international conference on learning and teaching in computing and engineering (pp. 62-69). IEEE
13. Csapó, G. (2019): *Placing event-action-based visual programming in the process of computer science education*. In: Acta Polytechnica Hungarica, 16(2), 35-57
14. Pluhár, Z., Gellér, B.: *International Informatic Challenge in Hungary*. In: Teaching and Learning in a Digital World: Proceedings of the 20th International Conference on Interactive Collaborative Learning. Berlin, Germany: Springer, (2018).pp. 425-435
15. Pluhár, Z., Torma, H., Törley, G. (2019): *Hallgatói teljesítményértékelés az algoritmikus gondolkodás tükrében*. In: InfoDidact 2018 (pp. 203-212)

## Források

1. Computer Science Teachers Association. (2019). Standards | Computer Science Teachers Association. Csteachers.Org. <https://www.csteachers.org/page/standards>
2. Ministerul Educației Naționale și Cercetării Științifice- Realizat de SIVCO Romania. (2006). [www.edu.ro](http://www.edu.ro) :: Programe școlare 2006-2015.
3. Valentina Dagiene. (2004). Task examples | [www.bebas.org](http://www.bebas.org) Bebras.Org. <https://www.bebas.org/examples.html>
4. ELTE IK T@T Labor. (2016). e-Hód | HÓDítsd meg a biteket! <http://e-hod.elte.hu/>



## 7. Mellékletek

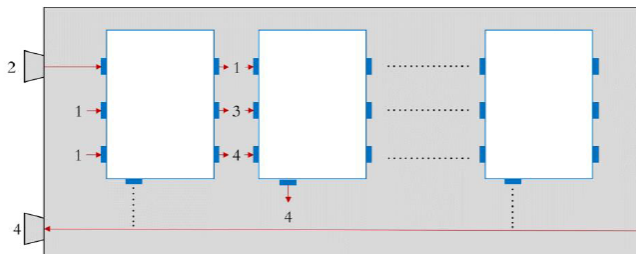
### 7.1. Számítógépes gondolkodás teszt – Számológép feladat

#### I. Számológép

A jelenkor tudósai építettek egy új számológépet, mely a következőképpen működik:

- Megadunk egy számot bemenetként és visszakapunk egy másikat
- A számológép belsejében kisebb számolóegységek dolgoznak. Míndegyik egység három számot (a, b és c) kap bemenetként és a következő lépéseket hajtja végre:
  - ha „a” egyenlő 1, akkor visszaadja a „c”-t mint eredményt
  - különben a következőket teszi
    - csökkenti „a”-t eggyel és az eredményt átadja a következő egység „a” bemenetére;
    - növeli „b”-t kettővel és az eredményt átadja a következő egység „b” bemenetére;
    - összeadja „c”-t a megnövelt „b”-vel, és az eredményt kiadja a következő egység „c” bemenetére.

Amikor a számológépnek megadunk egy számot, az az első számolóegység első (a) bemenetére kerül. A számolóegység másik két bemenete (b és c) pedig 1-et kap. Amint az egyik számolóegység eredményt ad vissza, ezt a számot kiadja a számológép, mint eredményt. A képen látható mi történik, amikor 2-t adunk meg a számológépnek. Ilyenkor két számolóegységet használ és 4-et ad vissza.



**Kérdések:**

1. Ha a számológépnek 4-et adunk bemenetként, milyen számot kapunk vissza?
  - a. 4
  - b. 10
  - c. 16
  - d. 20
2. Ha a számológépnek 7-et adunk bemenetként, milyen számot kapunk vissza?
  - a. 7
  - b. 21
  - c. 49
  - d. 70
3. Mi a következő művelet sor eredménye, ha a számológép(n) meghívás visszaadja a számológép által kiszámolt eredményt az n értékű bemenetre:  
 számológép(2) \* számológép(6) – számológép(3) \* számológép(4)
  - a. 0
  - b. 1
  - c. 10
  - d. 14
4. Általánosítás: mit ad vissza a számológép, ha N-et adunk bemenetként?  
 Eredmény: .....

**7.2. Számítógépes gondolkodás teszt – Hőtérvkép feladat**

**II. Hőtérvkép**

Egy gép a következő képeket, mint I, T, O, C és L betűket ismeri fel.



Mind az öt képhez egy „Különbségkártyát” állít elő. A különbségkártya az adott kép minden képpontjához egy értéket rendel. Az érték azt mutatja meg, hogy a többi kép esetében hánynál szerepel az adott helyen ugyanaz a képpont. A 0-s azt jelenti, hogy egy képnél sem szerepel ugyanazon a helyen ugyanaz a képpont, az 1-es érték azt jelenti, hogy egy képnél szerepel ugyanazon a helyen ugyanaz a képpont, a 2-es esetében két helyen, stb. Például az I betű esetében a következő különbségkártyát kapjuk:

0	3	1
1	1	3
1	4	1

**Kérdések:**

1. Melyik kép különbségkártyája a következő:

- a. T
- b. O
- c. C
- d. L

3	3	2
2	2	0
2	4	2

2. Melyik kép különbségkártyája a következő:

- a. T
- b. O
- c. C
- d. L

3	0	1
2	2	3
2	4	2

3. Melyik különbségkártya nem tartozik egyik betűhöz sem a felsoroltak közül?

a. 

3	3	2
1	1	3
1	4	1

b. 

3	3	1
2	2	3
2	4	2

c. 

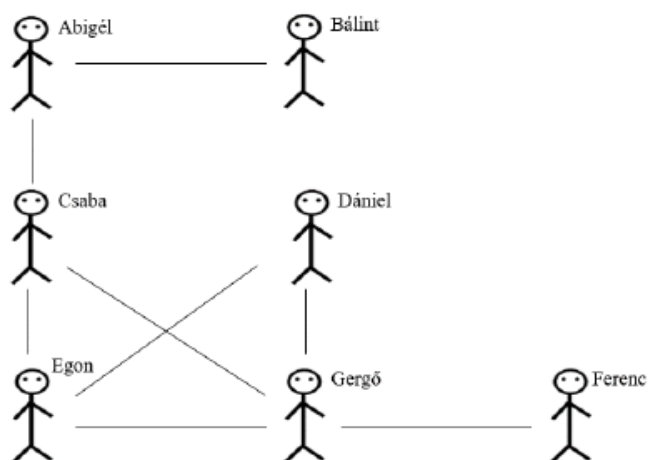
3	3	2
2	2	3
2	4	3

### 7.3. Számítógépes gondolkodás teszt – Népszerűség feladat

#### III. Népszerűség

A barátságok kezelésére több, jelenleg elterjedt szociális háló szolgál segítségül. Ezekkel lehetővé teszik például, hogy egy más oldalra írt bejegyzést több felhasználó lásson. Ezzel a lehetőséggel élve, hét kisdíák regisztrált egy online hálózatra.

Az alábbi kép, a hálózaton belüli baráti kapcsolatokat szemlélteti: a barátok egy vonallal vannak összekötve. Minden kisdíák egy-egy mottóval mutatkozott be, melyet megosztott a barátaival a hálózaton. Így a barátaik oldalán is megjelent a mottó. Minden kisdíák a saját oldalán és a barátai oldalán lévő mottókat látja.



**Kérdések:**

1. Kik láthatják a Dániel mottóját?
  - a. Egon és Gergő
  - b. Csaba, Ferenc, Egon és Gergő
  - c. Egon, Csaba, Abigél és Bálint
2. Hány olyan kisiák van, akiknek a mottóját pontosan 4 kisiák láthatja?
  - a. 3
  - b. 0
  - c. 4
3. Ha Csaba és Gergő barátsága felbomlik, és Csaba nem szeretné, ha Gergő továbbra is látná az bejegyzéseit, mit kell tennie?
  - a. Semmit, mert Csaba nem láthatja az üzeneteit
  - b. Fel kell bontania az Egonnal való barátságát
  - c. Új barátokat kell szereznie

**7.4. Számítógépes gondolkodás teszt – Megoldások**

**Answer Key**

- |      |      |      |       |
|------|------|------|-------|
| 1. c | 4. c | 7. c | 10. b |
| 2. c | 5. b | 8. b |       |
| 3. a | 6. d | 9. c |       |

## 7.5. Eredeti feladatok az e-hód online felületről

### 7.5.1. Eredeti Számológép feladat

HÓDÍTSD MEG A BITEKET - 2020-AS FELADATSOR

SZÁMOLÓGÉP (2020-DE-06A)

SZÁMOLÓGÉP (2020-DE-06A)

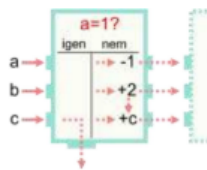
JUNIOR - NEHÉZ

SENIOR - KÖZEPES

A hódok építettek egy számológépet. Megadunk neki egy számot bemenetként és visszakapunk egy másikat.

A számológép belsejében kisebb számológységek dolgoznak. Mindegyik egység három számot (a, b és c) kap bemenetként és a következő lépéseket hajtja végre:

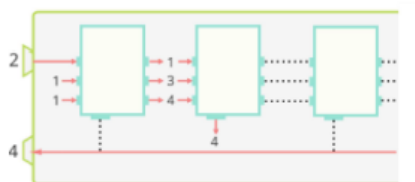
- ha „a” egyenlő 1, akkor visszaadja a „c”-t mint eredményt
- különben a következőket teszi
  - csökkenti „a”-t eggyel és az eredményt átadja a következő egység „a” bemenetére;
  - növeli „b”-t kettővel és az eredményt átadja a következő egység „b” bemenetére;
  - összeadja „c”-t a megnövelt „b”-vel, és az eredményt kiadja a következő egység „c” bemenetére.



Amikor a számológépnek megadunk egy számot, az az első számológység első (a) bemenetére kerül.

A számológység másik két bemenete (b és c) pedig 1-et kap. Amint az egyik számológység eredményt ad vissza, ezt a számot kiadja a számológép, mint eredményt.

A képen látható mi történik, amikor 2-t adunk meg a számológépnek. Ilyenkor két számológységet



használ és 4-et ad vissza.

**Ha a számológépnek 4-et adunk bemenetként, milyen számot kapunk vissza?**



## 7.5.2. Eredeti Hőtérkép feladat

HÓDÍTSD MEG A BITEKET - 2020-AS FELADATSOR

HŐTÉRKÉP (2020-DE-02)

HŐTÉRKÉP (2020-DE-02)

KADÉT - NEHÉZ

JUNIOR - KÖZEPES

SENIOR - KÖNNYŰ

Egy gép a következő képeket, mint I, T, O, C és L betűket ismeri fel.



Ehhez mind az öt képhez egy „Különbségkártyát” állít elő.

A különbségkártya az adott kép minden képpontjához egy színt rendel. A szín azt mutatja meg, hogy a többi kép esetében hánynál szerepel az adott helyen ugyanaz a képpont.

Minél világosabb egy szín a különbségkártyán, annál fontosabb az ezzel jelölt képpont a különbségek megadásánál.

Szín	Ennyi képnél szerepel ugyanazon a helyen ugyanaz a képpont
	Egy sem (0)
	1
	2
	3
	Mind (4)

Például a képnek ez lesz a különbségkártyája:

Melyik kép különbségkártyája a következő:

A) 	B) 	C) 	D) 
--------	--------	--------	--------



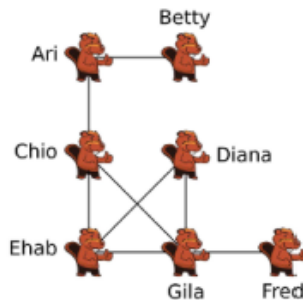
### 7.5.3. Eredeti Népszerűség feladat



HÓDítsd meg a biteket  
2015-ös e-HÓD feladatsor

**Népszerűség**  
(2015-CA-01)  
Junior - Nehéz, Senior - Közepes

Hét hód regisztrált egy online hálózatra. A kép azt mutatja, a hálózaton belül mely hódok "barátok": a barátok egy vonallal vannak összekötve. A nyári szünet után minden hód megosztott egy fotót a barátaival a hálózaton. Így a barátaik oldalán is megjelent a fénykép. Minden hód a saját oldalán és a barátaik oldalán lévő fotókat látja.



Kinek a fényképét látja a legtöbb hód?

A	Ari
B	Chio
C	Ehab
D	Gila





# PowerPoint prezentációk helyessége különböző korosztályokban

Papp Gréta<sup>1</sup>, Csernoch Mária<sup>2</sup>

<sup>1</sup>pappgreta997@gmail.com, <sup>2</sup>csernochmaria@inf.unideb.hu  
Debreceni Egyetem IK

**Absztrakt.** Kutatásaink során általános iskolások, középiskolások, felsőoktatásban tanulók és felnőtt előadók prezentációit vizsgáltuk meg. Arra kerestük a választ, hogy az egyes vizsgálati csoportok milyen megoldásokat alkalmaznak a prezentációk szerkesztésénél, hogyan használják a stílusokat, valamint azt, hogy a helyőrzőkön túl milyen további objektumokkal bővítik a diákat. Megvizsgáltuk továbbá a prezentációban előforduló sorok, szavak és karakterek számát, ezek eloszlását, és az egyes csoportok közötti eltéréseket. A csoportok közötti különbségeken túl, általánosan megfogalmazható, hogy az elrendezések közül a legtöbb esetben a Cím dia és a Cím és tartalom elrendezést használták helyesen, az 1–2 sorból álló diák mellett a 10–15 soros a legjellemzőbb. A felsőoktatásban tanulók használták a legkevesebb felesleges objektumot, míg a legtöbbet az általános és középiskolások. A prezentációk diáinak a számát tekintve tapasztalható eltérés az egyes csoportok között, ugyanakkor a szavak, a sorok és a karakterek számát vizsgálva nem találtunk szignifikáns eltéréseket.

**Kulcsszavak:** digitális szöveg, prezentáció, informatikaoktatás, PowerPoint, digitális írástudás, digitális nyelvhelyesség

*„Ha vannak ötleteink, gépek nélkül sem vagyunk elveszett emberek.  
Hiszen az ötletek birtokában a gépeket a magunk hasznárá hajthatjuk...  
A legtöbb ötletet egy bottal a homokba is berajzolhatjuk.” (Alan Kay) [1]*

## 1. Bevezetés

A digitális kultúra manapság még mindig csak kialakulóban van. Bár úgy tűnik egyre magabiztosabban használják a rendelkezésre álló szoftvereket, a végeredmény gyakran mégis telis-tele van hibákkal. Nemcsak a helyesírási – ugyanis a digitális helyesírást még mindig nem tanítják, ebből fakadóan sokan nem elég kritikusok ezekkel a szövegekkel –, hanem a szintaktikai, a szemantikai, a tipográfiai és a tördelési hibákra sem figyelnek kellőképpen oda. Nem alakult ki még a nagymértékű figyelem és elvárás arra, hogy ha közönség vagy olvasók elé készítünk digitális szövegeket, azok helyesen megszerkesztett, igényes, stílusos dokumentumok legyenek. Pedig sokkal több emberhez jutnak el ezek a szövegek, mint például egy kézzel teleírt füzet, ahol még arra is odafigyelnek az írni tanuló általános iskolás gyerekek, hogy pontosan egy mutatóujnyi helyet hagyjanak az új bekezdések első sora előtt; ahol a helyesírási hibákért pontot vonnak le; ahol figyelünk a küalakra, a címek kiemelésére, a szöveg megszerkesztettségére. A digitális szövegekkel – ellentétben a kézírásos szövegekkel – gyakran sokkal felületesebben bántanak a szerzők: nem figyelnek a bekezdésekre, nem használják megfelelően a nem nyomtatódó karaktereket, de a legfontosabb, amelyből a hibák nagy része fakad: nincs meg a munkafolyamat [2] tervezési fázisa és a három kulcsüzenet, amelyet át szeretnének adni a prezentációval [3, 28]. Éppen ezért a végeredmény kaotikus, összefüggéstelen hatást kelt, és a készítőjéről sem az igényesség és az átgondoltság tükröződik. A tanulmány elején kiemelt idézettel is az volt a

célunk, hogy az ötletelő, tervező fázisra hívjuk fel a figyelmet – hiszen „[a] szoftverek ugyan eltárolják a kész megoldásokat, de nem segítenek a létrehozásukban” [4, 102–103]. Kutatásaink során elsősorban a PowerPoint prezentációk tervezésére fókuszálunk. Célunk, hogy bemutassuk, milyen alapvető tervezési hibákat vétenek a prezentációk szerzői, milyen szempontokat hagynak figyelmen kívül, és ezáltal milyen hatást váltanak ki a közönségből.

### 1.1. A kutatás módszertana

A kutatás 120 PowerPoint prezentáció elsősorban kvantitatív elemzésére épül. Az összegyűjtött prezentációk egy része az interneten bárki számára elérhető prezentáció, másik része pedig különböző iskolákból, előadóktól érkezett. Az online gyűjtött diások miatt nem egyértelműen meghatározható a területi eloszlás, ezért ezt a szempontot figyelmen kívül hagytuk. A prezentációkat korosztály szerint négy kategóriába soroltuk:

- Általános iskola: olyan, legalább 4. osztályos tanulók prezentációi, amelyek készítői már tanultak informatikát, ismernek legalább egy szoftvert.
- Középiskola: ebbe a kategóriába szakiskolák, szakgimnáziumok és gimnáziumok diákjai tartoznak. A prezentációk jellemzően iskolai órákra, versenyekre készültek.
- Felsőoktatás: az egyetemisták kiselőadáshoz, versenyekhez készített előadásai.
- Előadói: ez a legtagabb kategória, amelybe TDK és konferenciaelőadások, valamint tanárok előadásai kerültek. Szintén ebbe a kategóriába tartoznak a különböző cégek prezentációi is.

Ennek a kategorizálásnak a célja, hogy megvizsgáljuk, az iskolai informatikaoktatás függvényében milyen tudáselemeket sikerült tárolni, ez hogyan jelenik meg a prezentációkon, tükrözi-e a diások helyessége, hogy egyre több informatikát tanultak a készítői, és van-e jellegzetes vonás, tipikus hiba 1-1 kategórián belül, vagy a kategóriák között.

Az egyes csoportokon belüli és közötti jellemzőknél megvizsgáltuk, hogy a szerzők milyen elrendezést használtak a diákon, ezek között milyen összefüggések fedezhetők fel. Az egy dián előforduló karakterek, sorok, szavak számát is megfigyeltük, hiszen ez az egyik legalapvetőbb szempont a prezentációk tartalmi elemzése során. Számos szabályt találhatunk arra, hogy hány sor, szó, fő gondolat lehet egy dián. Az egyik legjellegzetesebb az 1-7-7 szabály, amely szerint egy dián egy fő gondolat, maximum 7 sor és soronként maximum 7 szót használhatunk [4, 142]. Közös azonban ezekben a szabályokban, hogy a minimumra és a kulcsszavakra törekszik, hiszen George A. Miller kutatásában is megállapította, hogy az emberi agy munkamemóriája  $7 \pm 2$  információt képes befogadni [5, 81–97].

A helyesen szerkesztett prezentáció egyik további alapvetése, hogy a készítő dialrendezést – stílust – alkalmaz a hasonló tartalmak formázásához. Kutatásunk során azt is vizsgáltuk, hány esetben használtak a szerzők (helyes) diamintát, betartották-e a szó- és sorszámokra vonatkozó szabályokat. Bár ebben a cikkben nem térünk ki rá bővebben, a forráshasználatot, annak helyességét is elemeztük, illetve a fájlnevezésekre is kitértünk. A helyesírást kvalitatív módszerekkel vizsgáltuk, összefüggésbe hozva azzal, hogy a diákok milyen szabályokat sajátítanak el a nyelvtanórák során.

### 1.2. A digitális szövegek helyesírása a nyelvtan és a digitális kultúra órán

Helyesírás és szövegfelépítés (szövegszerkesztés) tanítása már egészen korán, alsó tagozatban megkezdődik. A helyesírást a teljes oktatási rendszerben nagyon komolyan veszik, már ami a kézzel írott szöveget illeti. Dolgozatnál, felvételinél és érettségien szigorúan veszik a hibákat, olykor még pontlevonás is jár értük. A helyesírás 3. osztálytól kezdve visszatérő témakör. Jelen kutatás során a 2020-ban bevezetésre került Nemzeti alaptantervet [6] vettük alapul, valamint az erre épülő kerettantervet az általános iskola 1–4 [7] és 5–8 osztályára [8], valamint a gimnázium 9–12. évfolyamára [9] vonatkozóan. A NAT 2020 alapján [6] *Digitális kultúra* néven tanulnak a diákok informatikát, által-

nos iskola 3. osztályától kezdődően egészen 11. osztályig. Az egyes témakörökhöz rendelt óraszámok változók, ezen belül a prezentációkészítésre szánt órák száma meglehetősen alacsony (**1. táblázat**).

	alapóraszám	prezentációkészítés
általános iskola 3–4.	68	0
általános iskola 5–6.	68	8
általános iskola 7–8.	68	6
középiskola 9–10.	102	0
középiskola 11.	68	0

**1. táblázat:** A *Digitális kultúra* tantárgy alapóraszámja és a prezentációkészítésre szánt órák a NAT 2020 alapján készült kerettantervek alapján.

Jelen tanulmány írásakor a 2020-as NAT-hoz készült *Digitális kultúra* tankönyv csak az 5. [10], 6. [11], 9. [12] és 10. [13] évfolyam számára készült el. Az eddig megjelent tankönyvek alapján az látható, hogy a digitális szövegek helyesírását egyik sem tanítja. Sokkal inkább az eszközcentrikus, formai megoldásokon van a hangsúly, amit prezentációkészítés alapjaiként mutatnak be a szerzők.

Bár nem sok órát szán a kerettanterv a prezentációkészítésre, számos fontos szabályt megemlít a szövegszerkesztés témakörnél, amelyek alkalmazhatók lennének a prezentációk készítésekor is. A két témakör összekapcsolása azonban elmarad, így izoláltan kerül oktatásra mind a szöveg-, mind a prezentációkezelés, teljesen figyelmen kívül hagyva a közös elemeket. A digitális szöveg értelmezésével, alapfogalmaival a magyar nyelvtankönyvek is foglalkoznak, bár nem fektetnek nagy hangsúlyt rá. Talán nem is gond, hiszen rengeteg alapvető definíciós hibát elkövetnek. A Kurucz István tankönyvfejlesztő által szerkesztett 10. osztályos nyelvtan munkafüzet [14] röviden foglalkozik a digitális szövegek helyesírásával, szabályaival. Bár ez sem túl részletes, de néhány fontos szabályt felmutat (idézőjel használata, kötőjelek, írásjelek előtti helytelen szóközpontok, ahogy azt az **1. ábra** mutatja), amit a tanárnak pontosítani és kiegészítenie is kell. Fontos azonban megjegyezni, hogy ez a tankönyv sem tesz említést arról, hogy ezek a szabályok mind megtalálhatók a Magyar helyesírás szabályaiban [15], és ezzel együtt a hivatkozások is hiányoznak. Mindezt figyelembe véve, teljesen indokolatlan egy majd 25 évvel korábban megjelent szövegszerkesztési tankönyvből idézni, amikor kiválóan használható a nyomtatott és online formában is elérhető, legfrissebb Magyar helyesírás szabályai [15]. A szövegszerkesztés könyvben [16] és a magyar nyelvtan munkafüzetben [14] említett szabályok a Magyar helyesírás szabályaiban az alábbi helyeken található meg sokkal részletesebben és pontosabban:

- Az írásjelek [15: 239–275.]
- A mondatokat záró írásjelek [15: 241.]
- zárójel, idézőjel, gondolatjel [15: 240.]
- kötőjel [15: 240., 263. 265., 275.]
- nagykötőjel [15: 264.]

Ugyanakkor óriási hiányossága a magyar nyelvtan munkafüzetnek [14], hogy nem tesz említést a digitális szövegekben alkalmazandó automatikus elválásról.

### „Helyesírási” szabályok

Ígazából a helyesíráshoz tartozik, de kézzel írott szövegek esetén nem olyan feltűnő azoknak a szabályoknak a be nem tartása, melyek a szavak és az írásjelek elhelyezését határozzák meg, emiatt ezeket a szabályokat kevesen ismerik jól. A legfontosabbak a következők:

- A mondatközi és mondatzáró írásjelek és az előttük álló szó között nincs köz, de a következő szó és az írásjel között van.

. , : ; ! ?

- A gondolatjel mindkét oldalán szóköz van, kivéve, ha tagmondat határára esik, és utána kettőspont, vessző vagy pontosvessző kerül.

– példa –

- A szón belül álló kötőjel (= elválasztójel) mindkét oldalán közvetlenül a szóhoz kötődik.

egy-két

- A zárójelek és a zárójelbe tett rész között nincs szóköz, a zárójelen kívüli rész és a zárójel közé – a mondatközi és a mondatzáró írásjelektől eltekintve – szóközt kell írni.
- Az idézőjelek használata a zárójelekével megegyező. Ügyeljünk arra is, hogy az idézőjelek („”) nem helyettesíthetők a " jellel (a coll jele), és az aposztrófok (') sem a ´ jellel (a perc jele).

Kézzel írott szöveg esetén majdnem egyformák az elkülönítésre, összekapcsolásra használt jelek, de nyomtatásban fontos megkülönböztetni ezeket. Ikerszavaknál és szóelválasztásnál az elválasztójelet használjuk, névpárok, számpárok esetén a nagykötőjelet, míg gondolat közbeelkelésére, párbeszédre a gondolatjelet.

(Ióth Tamás: *A szövegszerkesztés alapjai*. Nemzeti Tankönyvkiadó, Budapest, 1997. 28–29.)

1. ábra: Magyar nyelvtan munkafüzet 10. osztály [14, 67].

A 2020-as Nemzeti alaptanterv szerint úgy tűnik, a digitális szöveg helyessége a digitális kultúra órákat és a magyar nyelvtanórákat tartó tanár függvénye. Megdöbbentő azonban, hogy mindkét tantárgyból alacsony az óraszám, ráadásul a digitális szövegekről szóló fejezetek még mindig a peremterületet képezik a nyelvtanoktatásban. Bár fontos a kézírással született szövegek helyesírása, nagyobb hangsúlyt kellene fektetni a gépirásos szövegekre, hiszen egyre növekszik a digitális szövegek száma.

## 2. A prezentációk felépítése és szerkesztése

A helyesen szerkesztett digitális szöveggel, így a prezentációval szemben is követelmény, hogy a módosításokkal szemben invariáns legyen [17]. Ahhoz, hogy a hasonló tartalmú diáknak egységes megjelenítést adjunk és egyszerre lehessen módosítani ezek formázását, diamintát szükséges alkalmaznunk. A *Diaminta* nézetben ugyanis egyetlen kattintással képesek vagyunk bármilyen formázást módosítani, amely érvényes lesz az azonos elrendezésű diákon a teljes dokumentumon belül. Jelentős időt spórolunk meg ezzel, hiszen nem diáknaként vagy objektumként kattintgatva kell végrehajtunk a módosításokat, nem utolsó sorban a diák áttekinthetőbbek lesznek. A Nemzeti alaptantervhez készült – jelenleg elérhető – digitális kultúra tankönyvek megdöbbentően keveset foglalkoznak a diamintával. A tankönyv a bemutatókészítés tanítása során arra biztatja a diákokat, hogy különböző objektumokat – alakzatokat, rosszabb esetben szövegdobozokat – használva szerkesszék a prezentációkat [10].

A kutatásunk egy része arra irányult, hogy a kiválasztott prezentációkban megvizsgáljuk, milyen elrendezést használtak a szerzők, dolgoztak-e helyőrzőkkel, és hány extra objektum került egy-egy diára. Kategóriánként elemeztük azt is, hány diával dolgozott egy-egy szerző, és azokon hány sorból, szóból, karakterből áll a tartalom. Az eredmények alapján különböző minták rajzolódtak ki az egyes kategóriák esetében.

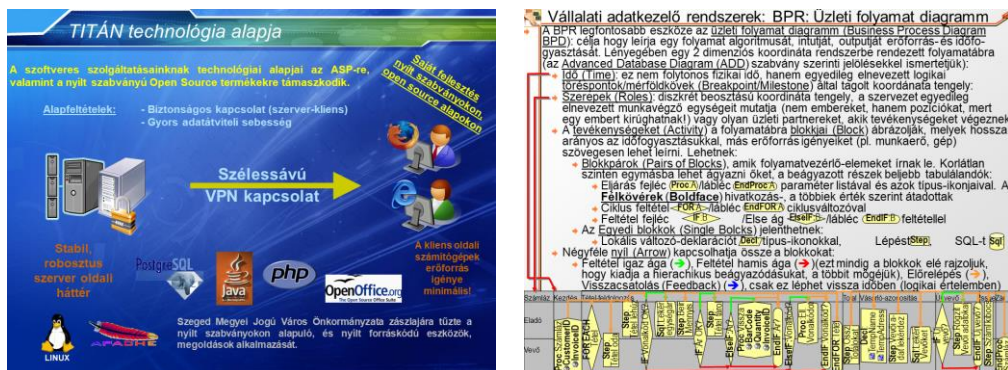
## 2.1. A prezentáció célja

Prezentációt többféle célból készítünk. Egy prezentáció elkészítését megelőzi a tervezés fázisa, mely során azzal is számolnunk kell, ki lesz a célközönségünk, milyen hosszú előadást tervezünk, és már ekkor gondolnunk kell arra is, hogy mennyire legyen „sűrű” a prezentáció. A különböző céloknak ugyanis különféle szerkezeti és grammatikai megoldásai lehetnek. A kutatás során elemzett 120 prezentációból három nagyobb kategória körvonalazódott. A három kategóriát elsősorban a grammatikai megoldások különítik el, másrészt az egy diára került szavak, karakterek száma, nem utolsó sorban pedig a vizuális megjelenés. A három típus tehát a következő:

- Előadáshoz készített prezentáció
- Tanuláshoz készített prezentáció
- Vázlatszerű prezentáció

### 2.1.1. Előadáshoz készített prezentáció

Ezek a diárok jellemzően színes, olykor túlgondolt és sűrű prezentációk. Mi ennek az oka? Sokan azt gondolják – tévesen –, hogy „minél több az információ, annál inkább azt tükrözi az előadásom, hogy felkészült vagyok”, hogy az illető „jó munkaeuró” [4, 142]. Hajlamosak az emberek megtölteni egyetlen diát végtelennek tűnő felsorolásokkal, túlmagyarázott mondatokkal vagy tagmondatokkal, különféle betűtípust és betűméretet választva, és feltűnő, színes háttereket, még rosszabb esetben élénk betűszínnel. Mindezt úgy, hogy gyakran megtörve a lineáris olvasást, döntött sorokat, szövegdobozokat használnak. Egy jó prezentáció titka éppen az ellenkező: letisztultság, minimalizmus, stílusos alkalmazása. Emellett természetesen a szintaktikai és szemantikai szempontokat sem szabad figyelmen kívül hagyni. A helyesírást ellenőrizni kell többször is. Kellemetlen helyzetet okozhat, ha hibákkal teli, túldekorált, túl sok információt tartalmazó lapokat tárunk a közönség elé. Egyszerűen komolytalaná válik az előadó.



2. ábra: Sűrű, átláthatatlan, elrendezési hibákkal teli diaképek

A 2. ábra mutatja, hogy a felesleges információkkal teli diaképek nagy zavart keltenek, nincs egyetlen fókuszja, és egyszerre sok inger éri a hallgatóságot. Az ilyen diaképek hátránya, hogy a célközönség az információhalom közül nem tudja kiválasztani a legfontosabbakat, ezáltal az előadó sem hagy nyomot a fejükben. Ez a két diakép is jól tükrözi, hogy a szerző valószínűleg nem tervezte meg a folyamatot, mert nem derül ki, pontosan mit akar közvetíteni. Igénytelennek hat, ráadásul elgondolkodtat, hogy egy előadásra azért megyünk-e, hogy olvassunk, vagy azért, hogy meghallgassuk az előadót. A bal oldali diakép szerzője elkövette azt a hibát, hogy mindenáron ki akart emelni mindent: élénk színeket használt, képekkel töltötte meg a diát és aláhúzást is használt. Az előbbi két megoldást bizonyos mértékig érdemes is használni, de semmiképp sem szabad halmozni, mert épp az ellenkezőjét váltja ki. Az aláhúzás pedig rossz taktika, hiszen ezzel az olvashatóságot rontjuk. A

jobb oldali diakép szerzője a szövegrány elforgatásával nehezítette az olvasást, illetve nagyon kicsi térközzel dolgozott, ami pedig az átláthatóságot, a dián való tájékozódást nehezíti. Ő is próbálkozott aláhúzással kiemelni a lényegét, de sikertelenül. A néző fókusza a színek miatt a lap alá irányul, viszont a szöveg itt sem olvasható könnyen. A teljes mondatok használata is nagy hiba, ugyanis mi szükség az előadóra, ha mindent a prezentáción akar a szerző elolvasatni?

### 2.1.2. Tanuláshoz készített prezentáció

A járványhelyzet óta jelentősen megnőtt az ilyen típusú prezentációk száma. Elsősorban a tanárok által készített prezentációk sorolhatók ide, amely abban tér el az előző kategóriától, hogy általában fehér háttérszínt használ, Times New Roman vagy Arial betűtípust, egy dián leginkább egyféle betűméret és betűtípus jellemző. Grammatikailag a teljes mondatok jellemzők ezekre a diasorokra, tehát margótól margóig futó sorok, telített lapok, amelyek leginkább egy e-bookra hasonlítanak. Vizuálisan szegényes, ezáltal nehezen megjegyezhető az információk egy ilyen dián. Ezt a típust hiba lenne prezentációnak nevezni, inkább digitalizált könyvhöz lehet hasonlítani, hiszen kiemelések nélkül, gyakran szerkesztés nélkül töltik ki a hosszú sorok az oldalakat. Tanuláshoz pontosan olyan, mint egy rövidített kötelező olvasmány. Ezek a „prezentációk” ugyanis kivonatai szakirodalmi szövegeknek, összefoglalója, rövidített változata egy hosszabb szövegnek. Viszont ez a kivonat is szövegszerű, diáról diára ki kell emelni a lényegét, tehát a kivonat kivonatát készítjük el, ha ilyen jellegű prezentációval van dolgunk. Mivel általában összefüggő mondatok alkotják, rengeteg felesleges szó kerül fel egy ilyen diasorra.

Az eszközkészletben kikeressük a kombi panel nevű adatbeviteli mezőt, majd beszurjuk az űrlapba. Ha az eszközkészlet automatikusan nem jelenik meg tervező nézetre váltásnál, az Eszközök/Testreszabás menüpont alatt pipáljuk ki az eszközkészlet jelölőnégyzetet.

A kombi panel beszurása után megjelenő dialógusablakban válasszuk ki a „Szeretném, ha a kombinált lista vezérlőelem megkeresné az adott értékeket a táblában vagy lekérdezésben” opciót, majd lépünk tovább.

Ez után mindig azt a táblát válasszuk ki, melyből a listában megjelenő értékeinket szeretnénk venni, valamint ha a táblát kiválasztottuk, válasszuk ki azt a mezőt, mely a listában megjelenjen, valamint a táblához tartozó kulcsmezőt. Például a Mit táncol kapcsolótábla a Táncsoportot köti össze a Táncokkal, tehát kell egy lista melyből kiválasztjuk az aktuális táncsoportot, azaz vesszük a Táncsoport táblát és kiválasztjuk a cs\_kód, valamint a név mezőket.

#### A könyv előállításának munkafolyamata

##### 1. Papírkészítés

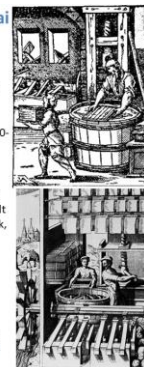
Már a könyvnyomtatás előtt évszázadokkal használnak papírt Európában → az első papírmalmok a mór (1140), majd a keresztény (1260) Spanyolországban működtek.

**Magyarországon** a külföldről behozott papír használata az 1310-es évektől kimutatható. 1530 előtt már működött nálunk papírmalom.

A papír (vagy papíros) alapanyaga: **rongy** → **péppé** verték → vízjégyes szítával **merítették** → **megszárlították** → **préselték** → **írássra** használták (fokozatosan kiszorította az állatdortól készült **pergament** az íráskészítésből, a levelezésből és az írott könyvek, a kódexek készítéséből). (Ld. jobbra fent a papírkészítőt a fametszeten.)

A papír kevésbé ellenálló és nem lehetett újra felhasználni (kikaparva a szöveget), mint a pergament. De előnye, hogy **olcsóbb és könnyen kezelhető, ill. állandó méretű**.

A papírgyártáshoz **víz kellett** (merítéshez, ill. hajtóerő a gyártási folyamatban: a papírmalmokat gyors folyású patakok, folyók mellé telepítették). (Ld. jobbra lent a papírmalmot a 17. századi rézmetszeten.)

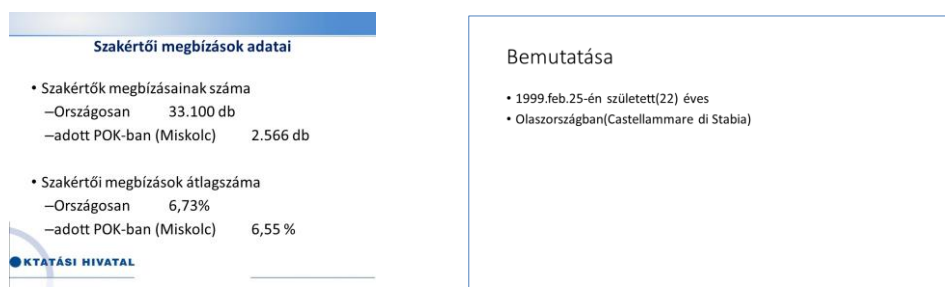


### 3. ábra: Szöveggel zsúfolt diaképek

A **3. ábra** arra bizonyíték, hogy egy teleírt diakép olyan, akár egy e-book. A bal oldali diaképen a szerző nem használt kiemelést, ezzel azt érte el, hogy nem kapott fókuszt a kép. A dia a hallgatóság számára csupán karakterhalom, hiszen, ha az előadóra figyel, esélytelen, hogy végigolvassa a dián lévő szöveget. Következésképpen teljesen felesleges ilyen diasorral készülni, mert senki sem fogja elolvasni. A jobb oldali ábra azt mutatja, hogy a szerző már használt valamiféle kiemelést, viszont nem jól választotta ki a lényegét: ha nem írta volna oda, hogy „Ld. jobb fent...”, a néző figyelmét enélkül is odavonzotta volna. A diaképen zavaró a sűrűség, nem ad teret az ürességnek, nem engedni „lélegezni” a hallgatóságot.

### 2.1.3. Vázlatszerű prezentáció

Ezek azok a diasorok, amelyeket címszavak, maximum rövid tagmondatok alkotnak, képet alig tartalmaznak. Ezt a típust azért különítjük el az elsőként említett (előadáshoz készített prezentáció) típustól, mert vizuálisan szegényes prezentációk, sokszor nem logikusak, inkább a mellérendelés, a felsorolás jellemző, az elrendezést tekintve monoton, egyhangú, és lineáris olvasást kínál.



4. ábra: Minimális diaképek

A 4. ábra a „minimális diaképek” címet kapta, amit a szó negatív értelmében használunk. A jobb oldali képen bár kevés szót használ a szerző, vizuálisan semmilyen hatást nem képes kiváltani ez az elrendezés a hallgatóságból. A szerző megreked a *tervező* fázisnál – amennyiben volt –, a formázás ezután következett volna. Ráadásul minél kevesebb a szöveg, annál inkább ráirányul a figyelem a helyesírási hibákra. Itt például rögtön kiszúrni, hogy az 1999. után hiányzik egy szóköz, ahogy a *febr.*, a *született* és az *Olaszországban* után is. A bal oldali képen szintén több helyesírási hibával találkozunk. Az automatikus felsorolás helyett hosszú kötőjeleket szúrt be a szerző, ráadásul a listaelemek első szava tapad a felsorolásjelhez. Sorkizárttá formázta a bekezdéseket, ezért a szavak közti tér aránytalan, nehezen olvasható.

A három típus közül egy sem alkalmas arra, hogy a prezentáció figyelemfelkeltő, izgalmas és követhető legyen, mind a végleteket tükrözi. A jó megoldás azonban épp e három halmaz metszetében található. Legfontosabb hívószó a stílus, amelyen az egész előadás nyugszik. Emellett fontos a minimalizmusra törekvés, és az, hogy szem előtt tartsuk: minden kivetített dia egy-egy kép, egy-egy vizuális hatás a közönségre. Épp ezért kell törekedni a harmóniára, hiszen a prezentáció akár egy művészeti alkotás, átgondolt, jól megtervezett, harmonikus és logikus. Ezzel a törekvéssel érdemes hozzájárulni a prezentációkészítéshez, különben a munkánk céltalan.

## 2.2. A megfelelő elrendezés

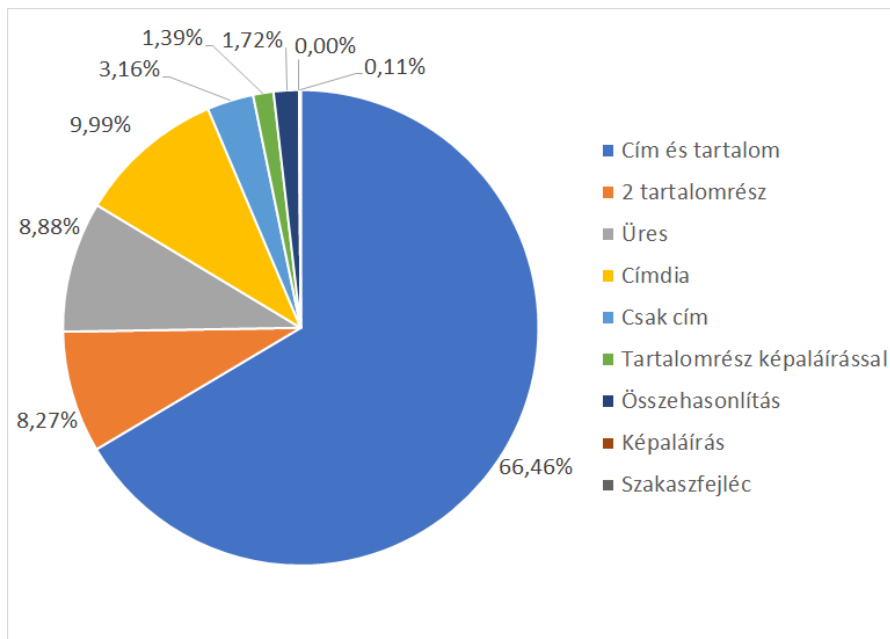
Mielőtt hozzájárulnánk egy diászor elkészítéséhez, több olyan fázisnak is meg kell előznie a felkészülést, amelyhez be sem kell kapcsolnunk a számítógépet. Garr Reynolds [18, 57–61] nyolc lépésben jut el a prezentáció elkészítéséig: első lépésben az ötletelésen van a hangsúly, ezután cél a nyugodt környezet megteremtése, ahol rá tudunk hangolódni a témára. Ez az úgynevezett „analóg üzemmód”: jegyzet-tömb, post-it, cédulázás. Ebben a szakaszban a hangsúly az ismeret bővítésén van. Ha elkészültek a jegyzetek, központi gondolatot kell keresni. Itt kell megtalálni a mondanivaló magját, amit közölni akarunk. Ez lesz az a gondolat, amit a prezentáció végig tükröz, és amire a hallgatóság emlékezni fog. Ha megvan a kulcsmondat, elő a táblával, papírral! Vázoljuk, csoportosítsuk az ötleteinket. Vázlatszerűen gondoljuk át, mit szeretnénk átadni a hallgatóságnak. Ezek után a tömörítés fázisához érünk, ahol a nagy ötletthalmazból kiemeljük a magot, a lényegét. Ha ezzel végeztünk, vázlatot készítünk papíron. Lerajzoljuk, hogyan képzeljük el az egyes „képeket” (= diákat, de még nem szoftverben gondolkodunk). Itt már törekednünk kell arra, hogy találjunk motívumokat, metaforákat, amelyek összefűzik a gondolatainkat. [18, 28] Utolsó lépésként válasszuk ki a megfelelő szoftvert, a vázlat alapján készítsük el a diamintáinkat, és ezt követően kezdhethünk hozzá az előadásunk tartalmának megszerkesztéséhez. Ezek a lépések lefedik a Pólya-féle koncepcióalapú problémamegoldási megközelítést [19], ahol Reynolds első hét lépése megfeleltethető Pólya első két lépésének, a nyolcadik lépése megegyezik Pólya harmadik lépésével (2. táblázat). Fontos azonban megjegyezni, hogy Reynolds leírásából hiányzik Pólya negyedik lépése, a diszkusszió, ami nem hagyható ki egy komplex problémamegoldási folyamatból.

	Reynolds lépései	Pólya lépései
1.	ötletelés	a probléma megértése, valamint az összefüggések megtalálása a meglévő és az új információk között
2.	zavaró tényezők kiküszöbölése	
3.	jegyzetelés	
4.	központi gondolat megfogalmazása	
5.	tudásbővítés	
6.	tömörítés, csoportosítás	
7.	vázlatkészítés papíron, diaképek megtervezése	
8.	vizuális anyag elkészítése szoftverben	a terv végrehajtása

**2. táblázat:** A Pólya-féle koncepció alapú problémamegoldási megközelítések és Reynolds-féle prezentációkészítés lépéseinek megfeleltetése

Reynolds prezentációkészítési javaslata jól hangzik, a valóság viszont nem ezt tükrözi. A szerzők nagy része az első hét lépést lelkiismeretfurdalás nélkül lépi át. Épp a legfontosabbat, hiszen, ha nincs ötletelés, nincs egyedi ötlet sem. Ha nincs vázlat, nincs mit csoportosítani. Ha nincs szakirodalmolvasás, nem tudunk eleget az adott témáról. És mindez nyomot hagy a prezentációnkon is.

Az általunk vizsgált 120 prezentációban az tükröződik, hogy a készítője nagyon kevés időt szánt a megtervezésre, hiszen nem használt megfelelő elrendezéseket, és teleírta a diákat szavakkal. Ezáltal az előadó maga sem tudja, mit is szeretne a hallgatóság felé közvetíteni. Az **5. ábra** mutatja, hogy a prezentáció készítői milyen elrendezéseket használtak.



**5. ábra:** Használt elrendezések



Legnépszerűbb elrendezés a *Cím és tartalom*. Ez valószínűleg azért a leggyakrabban használt elrendezés, mert a *Címdia* elrendezés után egy új dia beszúrásakor ezt az elrendezést rendeli alapértelmezésként a PowerPoint a beszúrt diához, mely ugyanakkor sablonos és unalmas lehet. Dobogós a *Címdia* és az *Üres* elrendezés, ahol a *Címdia* elrendezés esetén gyakran kerültek a diára a helyőrzőtől eltérő egyéb objektumok is. Ez összefüggésben állhat azzal, hogy az informatikaórákon a szövegdobozok és objektumok beszúrását preferálják – helytelenül (vö.: lakótornyos feladat az 5. osztályos *Digitális kultúra* tankönyvben [10, 48]). A felsorolt elrendezések alapértelmezettként vannak jelen a PowerPoint prezentációknál, de egyedi elrendezésmintákat is létrehozhatunk. Ha a stílusokat – elrendezéseket – használjuk, ez már lehet egy „motívuma” az előadásnak. Az elrendezések használata azonban igényli az átgondoltságot. Csak akkor tudunk mintázatot létrehozni, ha már előttünk van a terv, tehát az ötletelő, vázlatkészítő fázisra mindenképp szükség van.

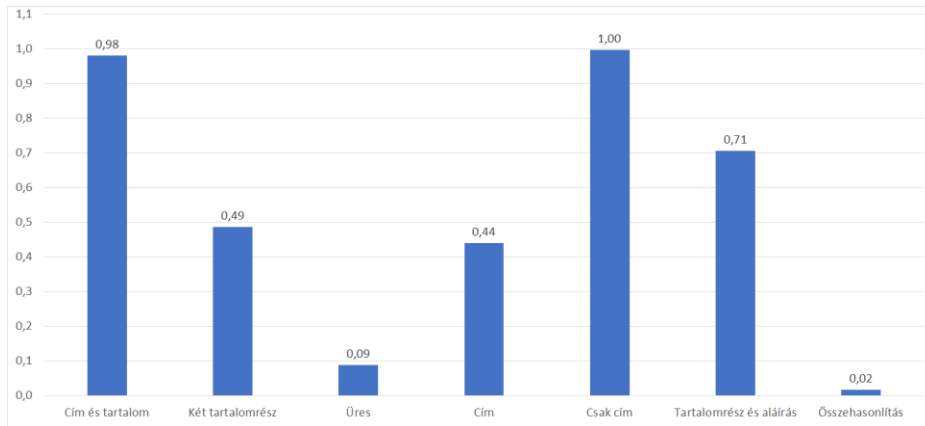
	Cím és tartalom	Címdia	Csak cím	Üres	2 tartalomrész
általános iskola	67%	10%	1%	9%	7%
középfiskola	57%	11%	3%	14%	10%
felsőoktatás	69%	13%	2%	4%	12%
előadói	70%	7%	5%	9%	6%

3. táblázat: Használt diaminták kategóriánként.

Kategóriánként vizsgálva a prezentációkat megállapítottuk, hogy a *Cím és tartalom* elrendezés a leggyakoribb (3. táblázat). Az általános iskola kategóriában a *Címdia* (10%) és az *Üres* elrendezés (9%), a középfiskolában az *Üres* (14%) és a *Címdia* (11%), a felsőoktatásban a *Címdia* és a *2 tartalomrész* (rendre 13% és 12%), az előadói kategóriában pedig az *Üres* (9%) és a *Címdia* (7%) elrendezés a dobogós. A konklúzió az, hogy az elrendezésre többnyire kevés időt fordítanak a készítő, a helyőrzők helyett újabb objektumok beszúrása mellett döntenek, és – jó esetben – *Üres* elrendezést használnak.

A 3. táblázat mutatja, hogy a középfiskolás prezentációk használják a legváltozatosabban az elrendezésmintákat (ha egyáltalán használnak). Ennek magyarázata, hogy a középfiskolai informatikaoktatásban megjelenik a prezentáció témakör, és egy gyakoribb a prezentációs szorgalmi és házi feladat. A vizsgált előadói prezentációknál egységesebb az elrendezésválasztás, ugyanis 70%-ban használnak *Cím és tartalom* elrendezést, amely az alapértelmezett a 2. diától. Gyakran előforduló hiba továbbá, hogy bár *Cím és tartalom* elrendezést használnak, de nem veszik figyelembe a helyőrzőket, hanem új objektumokat hoznak létre a szerzők. Érdekes tapasztalat, hogy új elrendezésmintát jellemzően nem hoznak létre egyik kategóriában sem, ha használnak az alapértelmezettől eltérőt, akkor a sablonok közül választottak.

Az elrendezésekhez tartoznak helyőrzők is, kivéve az *Üres* elrendezést. Ezeket a helyőrzőket gyakran nem használják ki a szerzők, hanem beszúrtak más-más objektumokat helyettük, ami azért gond, mert így egyesével kell beállítani a szövegre vonatkozó tulajdonságokat, ezáltal jóval időigényesebb a munka.



6. ábra: Korreláció a helyes és a használt elrendezések között.

A 6. ábra azt mutatja, hogy a szerző azt az elrendezést használja-e, amely a tartalomnak leginkább megfelel. Az elrendezéseket diánként vizsgáltuk, majd vetettük össze azzal, hogy a tartalomnak megfelelően milyen elrendezésmintát kellett volna használnia a szerzőnek. A 6. ábra elrendezésenként mutatja, hogy hány százalékos egyezést tapasztaltunk a tartalomnak megfelelő és a választott elrendezések között.

A *Cím és tartalom*, a *Csak cím* és – határeset, de még – a *Tartalomrész képaláírással* elrendezés esetében a szerzők többnyire jól választották meg az elrendezést. Közepes korreláció figyelhető meg a *2 tartalomrész*, és a *Cím dia* elrendezések esetében. Utóbbinál inkább a *Cím és tartalom*, és egy extra objektum beszúrásával érték el ugyanezt az elrendezést. Nagyon alacsony a korreláció az *Üres* és az *Összehasonlítás* elrendezések esetén. Az *Összehasonlítás* elrendezést bár 31 diánál választották, 33-nál lett volna erre szükség. A korreláció azért ennyire alacsony, mert ahol használták, ott sem helyesen tették, pedig sok esetben praktikus, szemléletes és logikus elrendezést mutat, átlátható, és könnyebben módosítható a stílusuk, mint a beszúrt objektumok használata esetén. Ebben az esetben fordul elő a leggyakrabban, hogy különféle szövegdobozok és alakzatok használatával „barkácsolják” össze azt, amit a helyesen megválasztott elrendezéssel egy kattintással megkaphattak volna. Ebből is látszik, mennyivel energiatakarékosabb az elrendezések használata.

### 2.3. Diák száma az egyes prezentációkban

Konferenciákon nem meglepő, ha előadásonként 30–40, vagy akár több diából álló prezentációval találkozunk. Ettől nemhogy kíváncsiak nem leszünk, aki teheti, ki is fordul a teremből. Várhatóan unalmas, hosszú és tömény előadásokra lehet számítani ezek alapján. Azt, hogy egy több tíz diából álló prezentáció mennyi időre lett tervezve, nehéz megmondani. Attól függ, mennyi időt beszél az előadó 1-1 diáról. De valóban tükrözi ez az előadás minőségét? Szerzőként hajlamosak vagyunk azt gondolni, hogy ha teleírjuk szöveggel a diákat, beillesztünk egy halom képet, és még animáljuk is, akkor biztosan jó munkaerőnek, szorgalmas kollégának gondolnak majd bennünket, és látványos, szórakoztató előadást tarthatunk. Tévedés! A hatásos, stílusos előadás kulcsszavai az egyszerűség, visszafogottság és természetesség [4, 19]. Az elemzett prezentációk szerzőinek az elképzelése ezzel épp ellentétben áll. Az elemzett prezentációk is azt tükrözik, hogy ezzel a szabállyal sokan nincsenek tisztában, és a legnagyobb gond, hogy a tankönyvek sem törekednek arra, hogy a letisztultságra ösztönözzék a diákokat. A vizsgált 120 prezentációban is megjelennek nagyon sok diából álló prezentációk, a határt viszont nem érik el, hiszen az itt mért maximális számú diát tartalmazó előadás még mindig nem a legrosszabb, amit egy szerző képes elkövetni. A vizsgálat eredményeit a 4. táblázat mutatja.

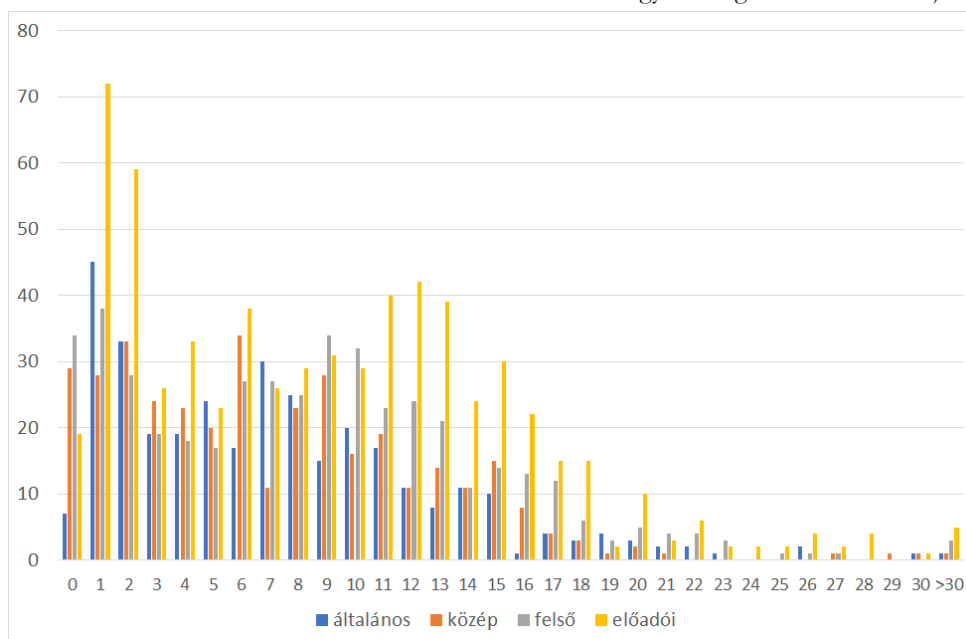
	minimum	maximum	átlag	medián	módusz	szórás
általános iskola	4	26	11,2	9,5	7	6,44
középfiskola	3	35	11,8	9,5	6	7,668
felsőoktatás	5	42	15,0	13	14	7,799
előadói	8	46	21,9	20,5	11	9,395

4. táblázat: Az egy prezentáción belüli diák számának leíró statisztikája kategóriánként.

A 120 prezentációt vizsgálva, a legkevesebb diaszám egy prezentációban 3, a legtöbb 46. Átlagosan 22 diát tartalmaz egy-egy előadás. A diák számából nehezen következtetünk arra, melyik szerző milyen hosszú előadást tervezett, hiszen egy 4–5 diából álló prezentáció is segíthet egy többórás előadást, éppúgy, ahogy egy 40 diából álló prezentációt is használhatunk egy félórás előadáshoz. Kérdés csupán az egy diára jutó idő.

A terjedelem fokozatosan nő, de nem ugrásszerűen. Természetesen biztosan léteznek akár 100 oldalas előadói prezentációk is, viszont a vizsgálat során nem dolgoztunk ilyen terjedelművel, ami a véletlen műve. Bármiféle megkötés nélkül gyűjtöttük a prezentációkat. Könnyen előfordulhat, hogy ha nagyobb mennyiségű szöveggel dolgozunk, a módusz nagyobb eltérést mutatna az egyes kategóriák között.

A diákon a szavak és objektumok számát a lehető legkevesebbre érdemes csökkenteni, és helyet engedni az „üres térnek” [4, 157]. A szöveget tekintve a lehető legkevesebb szövegnek kell a diára felkerülnie, és a lehető legnagyobb betűtípussal. A cél a vizuális hatás, ugyanis minden egyes kép kivált valamilyen hatást a nézőből. Különböző szabályokat találunk arról, hogy egy diára hány szó, sor vagy fontos gondolat kerüljön. Ahogyan korábban említettük a legelterjedtebb az 1-7-7 szabály [4, 142], de vannak, akik a  $6 \times 6$ , vagyis 6 sor soronként 6 szónál húzzák meg a határt [2, 29]. Átlagban tehát 36–49 szó. Az elemzett diákon lévő sorok számának gyakoriságát a 7. ábra mutatja.

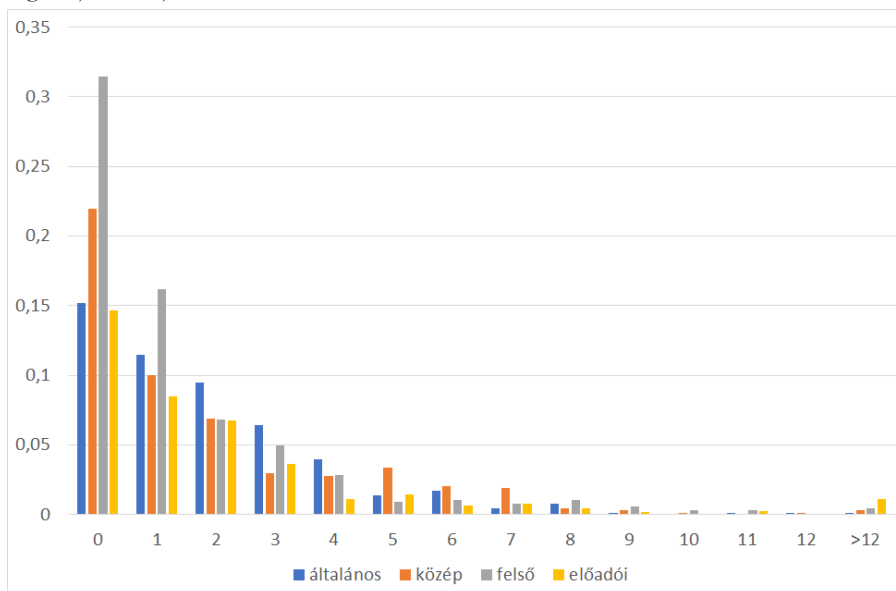


7. ábra: Sorok számának gyakorisága.

A prezentációt készítők közel egy harmada tartotta be az 1-7-7 szabályt [4, 142]. Már ebből egyértelműen látható, hogy sok esetben túlszűfolt, szöveggel (és/vagy objektumokkal teli) diákat használnak. A 7. ábrán mutatja, hogy leggyakrabban 0–15 sort használnak a szerzők, de előfordult az is, hogy 30-nál több sor került egyetlen diára. Ha jól látható, jól olvasható betűmérettel dolgozunk, akkor belátható, hogy 7 sornál többet nem érdemes írni egy diára, főleg, ha képeket vagy táblázatot, ábrákat is használunk. Egyértelmű, hogy az előadói prezentációk oszlopa kiugróan magas, ez annak is következménye, hogy több diából álló prezentációkról van szó. Amennyire magas számot kapott az egysoros diák száma, annyira magas a 7+ sort tartalmazó diák száma, és a legmagasabban végig az előadói oszlop áll. Tehát kimondható, hogy a legsűrűbb prezentációk az előadói kategóriából származnak.

A jó arányérzék ebben is fontos, ugyanaz igaz erre az esetre is, ahogy azt a bevezetőben már említettünk: a kevesebb több, és ne könyvet tárjunk a közönség elé, miközben azt szeretnénk elérni, hogy odafigyeljenek arra (is), amit előadóként mondunk. Szem előtt kell tartanunk, hogy a diasor nem helyettünk beszél, hanem szerkezetet ad a mondandónknak. Ha pedig csak 30 (vagy annál több) sorban tudjuk leírni a lényegét, akkor igazából nem is értjük, mit akarunk vele közölni, épp ezért fontos leszögezni a tervezés folyamatánál, mi a célunk, mert ha tisztában vagyunk szándékainkkal, 1 vagy 2 mondatban tökéletesen össze tudjuk foglalni a lényegét.

Arra, hogy egy diára maximum hány képet illesszünk be, nincs szabály, de ha logikusan belegondolunk, a kevesebb itt is több [4, 160]. A sok vizuális hatás sem segíti a befogadónak az értelmezést, terhelte teszi az előadást. A legkényelmesebb az egy fotó diánként, és fontos, hogy ezek is jó minőségűek legyenek. Objektumok esetében nem biztos, hogy szükség van maximalizálásra, hiszen nehezen lehet megállapítani, hogy hol a felső határ, különösen abban az esetben, ha animáció is párosul a grafikai elemekhez. Az elemzett diasorokban még nem került sor az objektumok alaposabb vizsgálatára, jelenleg a helyőrzőkön kívüli extra objektumok számát vizsgáltuk, ebbe beletartoznak a képek, táblázatok, alakzatok stb. Az elemzett 120 prezentációban szereplő extra objektumok eloszlását a **8. ábra** diagramja mutatja.



**8. ábra:** Felesleges objektumok relatív gyakorisága.

A szerzők itt sem követték azt a szabályt, hogy a kevesebb több. A legtöbb képet a középiskolás tanulók prezentációi tartalmazzák, a legkevesebbet pedig a felsőoktatásban tanulók diasorai. Kérdés, hogy van-e értelme ennyire terhelte tenni a diákat. A túl sok információ között eltűnik a lényeg, és a kiindulásként megfogalmazott célkitűzést sem sikerül megvalósítani: a hallgatóság semmit nem fog megjegyezni az előadásunkból.

### 3. Összegzés

Jelen tanulmány 120 prezentáció elemzését mutatja be négy kategóriába sorolva: általános iskola, középiskola, felsőoktatás és előadói.

Megállapítottuk, hogy a legtöbb probléma az előadói prezentációkkal van, annak ellenére, hogy az ebbe a kategóriába sorolt prezentációk készítői felnőttek, akik feltehetően a többi korcsoporttal szemben a legtöbb informatikaórán vettek részt. Sem a diák számát, sem a szavak és a sorok számát nem tartották szem előtt a szakirodalmi javaslatoknak megfelelően, a minimalizmus helyett a maximális térkihasználtságra törekedtek, figyelmen kívül hagyva az „üres”, fehér részeket, aminek fontos szerepe lenne egy prezentációban. Az általános iskolás diákok többnyire kevés diából álló prezentációt készítettek, ami összefüggésben lehet azzal, hogy a tankönyv feladatai is csak néhány diából álló diasort várnak el a diákoktól, nem ritka az egyetlen diát tartalmazó prezentáció sem. A középiskolás tanulók és az egyetemisták diasorai között nem volt túl nagy eltérés, a középiskolások az informatikaórán tanult formázások egy részét alkalmazzák, de érzékelhető, hogy a legfontosabb szempontokat mégsem veszik figyelembe.

A tankönyvek feladatai sem arra készítik fel a diákokat, hogy „helyes prezentációt”, azaz könnyen módosítható diasort hozzanak létre, inkább nem nyomtatható karakterekkel és objektumokkal érik el – bár hosszabb idő alatt – ugyanazt a megjelenést, amelyet egyébként stílusokkal – diaminta elrendezésekkel – sokkal egyszerűbb lenne létrehozni.

A konklúzió tehát, hogy a Nemzeti alaptanterv és a kerettantervek nem biztosítanak elég időt arra, hogy a prezentációelemzést és -tervezést alaposan megtaníthassák a tanárok a diákoknak. Sokkal inkább az eszközhasználaton van a hangsúly. Ebből következik, hogy a diákok tudása felületes, a tudás a rövid távú memóriáig jut el, és felnőtt korban már nincs igény a jól megformált diasorok elkészítésére. Nagy gondot jelent továbbá, hogy a prezentációkészítés folyamatát nem hangsúlyozzák a tankönyvek, a diákoknak így nincs is megtanítva, mi a prezentációkészítés lépéseinek helyes sorrendje, valamint az sem, hogy hogyan kell hozzájárítani egy diasor elkészítéséhez.

A digitális szövegek helyességéről – amelyről ebben a dolgozatban kevésbé esett szó – még mindig nem tanulnak eleget a diákok. Úgy gondoljuk, a digitális szövegek helyességének tanításához kevés az 1-2 nyelvtanfeladat 10. osztályban vagy néhány prezentációs eszközre koncentráló informatikaóra. A digitális szövegek helyesírását még inkább hangsúlyozni kellene, hiszen manapság ezek vesznek körül bennünket. Érdemes lenne felhívni a figyelmet arra is, hogy a közösségi oldalakon elterjedt az ékezet nélküli írásmód, ami nem egy biztató tendencia. Ez is jól mutatja, hogy a digitális szövegek helyesírására egyelőre még nem helyeződött akkora figyelem, nem vált még elvárássá, hogy a nyilvánosan megjelent szövegeket – akár egy Facebook-hozzászólást is – igényesen, jól megformálva, helyesen írjunk le. Ezek a szövegek nagyobb nyilvánosságot kapnak, évtizedek múlva is olvasható lesz 1-1 cikkünk, prezentációnk, így fokozottabban oda kellene figyelniük arra, amit leírunk. Ha már nyomot hagyunk a digitális térben, legalább helyesen megformált, helyesen írt szöveggel tegyük azt.

## Irodalomjegyzék

1. J. Schuster: *A bicycle for the mind – interview with Alan Kay, inventor of the Graphical User interface, Electronic Learning*, April 1994.  
[https://marcusdenker.de/old/\\_A\\_bicycle\\_for\\_the\\_mind\\_.html](https://marcusdenker.de/old/_A_bicycle_for_the_mind_.html) (utoljára megtekintve: 2021. 11. 09.)
2. C. Atkinson: *Hatásos prezentáció*, ford. Domoszlai László et al., Szak Kiadó, Bicske, 2008.
3. C. Gallo: *Steve Jobs a prezentáció mestere*, ford. Kuntner Gábor, HVG Kiadó Zrt., Budapest, 2010.
4. G. Reynolds: *PreZENTáció*, ford. Nagy Marcell, Budapest, HVG Könyvek, 2009.
5. G. A. Miller: *A varázslatos hetes szám, plusz vagy mínusz kettő: Bizonyos korlátok vannak az információfeldolgozás képességére nézve*, Pszichológiai Szemle. 1956, 63/2, 81–97.
6. Nemzeti Alapntanterv 2020.  
[https://www.oktatas.hu/koznevelas/kerettantervek/2020\\_nat](https://www.oktatas.hu/koznevelas/kerettantervek/2020_nat) (utoljára megtekintve: 2021. 11. 01.)
7. *Kerettanterv általános iskola 1-4 évfolyama számára*.  
[https://www.oktatas.hu/koznevelas/kerettantervek/2020\\_nat/kerettanterv\\_alt\\_isk\\_1\\_4\\_evf](https://www.oktatas.hu/koznevelas/kerettantervek/2020_nat/kerettanterv_alt_isk_1_4_evf) (utoljára megtekintve: 2021. 11. 10.)
8. *Kerettanterv az általános iskola 5–8 évfolyamára*.  
[https://www.oktatas.hu/koznevelas/kerettantervek/2020\\_nat/kerettanterv\\_alt\\_isk\\_5\\_8](https://www.oktatas.hu/koznevelas/kerettantervek/2020_nat/kerettanterv_alt_isk_5_8) (utoljára megtekintve: 2021. 11. 10.)
9. *Kerettanterv a gimnáziumok 9–12. évfolyamára*.  
[https://www.oktatas.hu/koznevelas/kerettantervek/2020\\_nat/kerettanterv\\_gimn\\_9\\_12\\_evf](https://www.oktatas.hu/koznevelas/kerettantervek/2020_nat/kerettanterv_gimn_9_12_evf) (utoljára megtekintve: 2021. 11. 10.)
10. Lénárd A., Abonyi-Tóth A., Turzó-Sovák N., Varga P.: *Digitális kultúra 5*. Oktatási Hivatal. (2020)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG05TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG05TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
11. Abonyi-Tóth A., Farkas Cs., Turzó-Sovák N., Varga P.: *Digitális kultúra 6*. Oktatási Hivatal (2020)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG06TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG06TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
12. Varga P., Jeneiné Horváth K., Reményi Z., Farkas Cs., Takács I, Siegler G., Abonyi-Tóth A.: *Digitális kultúra 9*. Oktatási Hivatal (2020)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG09TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG09TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
13. Abonyi-Tóth A., Farkas Cs., Jeneiné Horváth K., Reményi Z., Tóth T., Varga P.: *Digitális kultúra 10*. Oktatási Hivatal. (2020)  
[https://www.tankonyvkatalogus.hu/pdf/OH-DIG10TA\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-DIG10TA__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
14. Kurucz István, *Magyar nyelv munkafüzet 10.*, Oktatási Hivatal (2020).  
[https://www.tankonyvkatalogus.hu/pdf/OH-MNY10MAB\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/OH-MNY10MAB__teljes.pdf) (utoljára megtekintve: 2021. 11. 01.)
15. *A magyar helyesírás szabályai, Tizenkettedik Kiadás*, Akadémiai Kiadó, Budapest, 2015.  
<https://helyesiras.mta.hu/helyesiras/default/akh12> (utoljára megtekintve: 2021. 11. 01.)
16. T. Tóth: *A szövegszerkesztés alapjai*. Nemzeti Tankönyvkiadó, Budapest, 1997.
17. M. Csernoch: *Teaching word processing – the theory behind*, In: Teaching Mathematics and Computer Science 7/1, 2009.
18. G. Reynolds: *A meztelen előadó*, ford. Kőrös László, HVG Kiadó Kft, Budapest, 2011.
19. G. Pólya: *How to Solve It*. Princeton University Press, 1945.







# Interdiszciplináris Informatikai Tehetségek

Sarmasági Pál

psarmasagi@inf.elte.hu  
ELTE IK

**Absztrakt.** Az új Nemzeti Alaptanterv (NAT2020) szorgalmazza a digitális kultúra alkalmazását a különböző tantárgyak tanulása során. Ez sok esetben segíthet bevonni a más tárgyra fókuszáló diákokat a digitális kultúra tantárgy tanulásába, ha sikerül felismertetni velük a digitális kultúra alkalmazásának előnyeit az adott területen. Jelen cikk célja személyes életutak példáján keresztül bemutatni, hogy az informatikai ismeret milyen szakmai előnyöket kínál nem informatikai területen dolgozók számára. Nyelvtan, orvos, gyógyszerész vagy éppen pénzügyi szakember életében, munkájában is komoly karrier előnyt jelentett a digitális kultúrában való jártasság, a programozási alapismeretek megszerzésével kialakított algoritmikus gondolkodás. A példák motiválják a diákokat, támogatva a digitális kultúra minden tantárgyra kiterjedő alkalmazásának tantervi célját.

**Kulcsszavak:** Informatikai tehetség, interdiszciplináris, algoritmikus gondolkodás, motiváció

## 1. Bevezető

Az informatikai bennszülöttek korábban sokan már kisiskolás koruktól kezdve okostelefont használnak a mindennapokban. A szórakoztató videók megtekintése, a számítógépes játékok vagy a közösségi média kezelése ugyan rutinos eszközhasználatot eredményez, de nem vezet az algoritmikus gondolkodás kialakulásához. Belépő szintnek, kedvesnialónak hasznosak ezek az alkalmazások is, így az oktatás feladata, hogy a szórakozás felől elindulva megismertesse a diákokkal a számítógépet, mint problémamegoldó eszközt.

A legújabb alaptanterv, a NAT2020 új megközelítésben, a digitális kultúrának nevezett tantárgyban támogatja ezt a folyamatot. [1] A lexikális tudásról a problémamegoldás felé teszi át a hangsúlyt, a diákok számára is ismert, életből vett érdekes feladatok megoldásával kívánja fejleszteni az informatikai eszközök tudatos használatát. [2]

A diákok többségének középiskolás korban már kialakult attitűdje van az informatikával kapcsolatban, ami befolyásolja pályaválasztásukat is. Szép számmal vannak az informatika iránt elkötelezett diákok, akik szeretik az informatikát, nem csupán játszanak a számítógéppel, hanem érdeklődnek a működése iránt is. Ezen tanulók motiváltak az algoritmusok megismerésére, örömmel ismerkednek a programozási nyelvekkel, abban az esetben is, ha nem informatikai pályára készülnek. Mások jó eredményeikkel megelégedve, eldöntött pályaelektiójukra hivatkozva nem akarnak elmélyedni az algoritmusok világában. Végül vannak olyan diákok, akik többnyire valamilyen kudarcélményük miatt elzárkóznak az informatikatanulástól, megelégednek a közösségi média és a szórakoztató alkalmazások használatával. A kevésbé motivált diákok bevonása az órai munkába komoly kihívás minden tanár számára. Kutatásom egyik célja olyan motivációs lehetőségek keresése, melyek segítségével ezen diákok is meggyőzhetőek a digitális kultúra tanulásának fontosságáról, előnyeiről.

Az új tanterv szorgalmazza a digitális kultúra alkalmazását a különböző tantárgyak tanulása során. Ez sok esetben segít bevonni a más tárgyra fókuszáló diákokat, ha sikerül felismertetni velük a digitális kultúra alkalmazásának előnyeit az adott területen. A motiváció felkeltésére a tanterv csak bő keretet biztosít, a legtöbb tantárgy esetén – ami egyébként nagyon fontos – az internetes források keresését és használatát javasolja. A különböző tantárgyak és továbbtanulási lehetőségek ennél széle-

sebb körben építhetnek a digitális kultúra tantárgy témaköreire, mind az alkalmazások, mind a programozás, algoritmizálás tekintetében. A konkrét tantervek és tanmenetek kidolgozása még folyamatban van, a digitális kultúra oktatásának ez a bemeneti oldala. Érdeemes mintát venni a rendelkezésre álló kimenetből, az elmúlt években, évtizedekben végzett diákok munkahelyi tapasztalatát felmérve rámutatni, mely informatikai ismeretek, gondolkodásmódok segítik az emberek munkáját és előmenetelét a munka világában. Jelen cikk célja személyes életutak példáján keresztül bemutatni, hogy a digitális kultúra, az informatikai ismeretek magasabb szintje milyen szakmai előnyöket kínál a munkaerőpiacon, elősegítve ezzel a diákok motivációját, támogatva a digitális kultúra minden tantárgyra kiterjedő alkalmazásának tantervi célját. A digitális kultúra magas szintű alkalmazási készségére, az informatikai tehetségekre nem csak a professzionális informatika területén van szükség, hanem az élet és a tudomány szinte minden területén.

## 2. Történelmi kitekintés

Érdeemes felidézni, hogy a digitális kultúrát, a számítógépes gondolkodás szükségességét már az informatika hőskorában, az 1960-as években felismerték. A magyar származású Kemény János az általa vezetett Dartmouth Egyetemen arra törekedett, hogy az egyetem minden hallgatóját megismertesse a számítógéppel, annak kezelésével, programozásával. Azokban az időkben kevés számítógép volt, a meglévő gépek jobb kihasználására javasolta az időosztás alkalmazását, amivel egyidőben több felhasználó csatlakozhatott a számítógéphez. Kemény János és munkatársa Thomas Kurtz megalkotta a BASIC programozási nyelvet, hogy segítségével minden egyetemi hallgató megismerhesse a programozást. Az elnevezés egy rövidítés, ami ebben a formában is értelmes, mivel a „kezdő”, az „alap”, „alapozó” és „alapfokú” szavakkal is fordítható. A rövidítés kifejtése egyértelműen utal a programnyelv céljára: Beginner's All-purpose Symbolic Instruction Code, azaz kezdőknek való, általános célú szimbolikus utasításkód. Lényeges jellemzője volt a korai BASIC programozási környezeteknek az értelmezős (interpreter) megvalósítás, ami interaktivitásával támogatta az ismerkedést a számítástudománnyal. Kemény János a Dartmouth Collage rektoraként az 1980-as évek végére elérte, hogy végzős diákjainak 90%-a digitális írástudó lett. Bölcsész, jogász és közgazdász hallgatók nem csupán szövegszerkesztésre vagy játékokra használták a számítógépet, hanem programozni is megtanultak.[3]

Az elmúlt 30 évben sokat fejlődött az informatika, a digitális írástudás a középiskolában közel 100%-os. Az okostelefonok elterjedése mellett az infokommunikációs eszközök felhasználó szintű kezelése már általános, az email kezelés, a szövegszerkesztő alkalmazások és a webböngészők alapszintű használata nem okoz gondot a középiskolás diákoknak. Az eszközök tudatos használata, a nem rutinszerű használatból eredő problémamegoldás már kevésbé elterjedt ebben a korosztályban. Az algoritmizálási és programozási ismeretek terén pedig a diákok nagyon szerény csoportja rendelkezik a szükséges gyakorlattal. A digitális kultúra oktatása így továbbra is indokolt, amit az eszközök fejlődése és az életünkre gyakorolt növekvő hatása még indokoltabbá tesz.

## 3. Kutatási cél és módszertan

A középiskolás diákok a felnőtt kor felé közeledve gyakorlással fejlesztik vitakészségüket. [4] Ennek gyakori területe a tanórán megvitatni a tanárral, miért van, vagy éppen miért nincs szükségük az adott tananyag megtanulására. Az informatika tanórákon sokszor hangoztatott érv egyike, hogy egész nap a gép előtt ül, mindent tud, ami érdekli. Másik gyakori érv szerint olyan területen szeretne továbbtanulni, ahol csak minimális szinten van szüksége a digitális kultúra alkalmazására. Ehhez hasonló érvelések az informatikában kimondottan tehetséges diákok részéről is hallhatók, akik nem ismerték még fel, milyen komoly versenyelőnyt jelent digitális kultúrában való jártasságuk. Egyrészt

ennek bemutatására, másrészt a digitális kultúrához való alkalmazkodás, tanulás melletti érveléshez konkrét személyes példákat is felhasználhatnak a tanárok. [5]

Elsődleges kutatási célom egy kvalitatív kutatás készítése volt, mely során olyan személyekkel terveztem interjút készíteni, akik nem informatikai vonalon tanultak tovább, de választott szakterületükön megalapozott digitális kultúrájuknak, tehetségüknek köszönhetően tudtak eredményes karriert építeni. Mik motiválták őket az informatikatanulásra és melyek azok a közös pontok, amelyek munkájuk során előnyt jelentettek számukra? Az ő történeteik, példáik erősíthetik a diákok motivációját a digitális kultúra tantárgy tanulására, a pályaeorientációjuknak megfelelő alkalmazások és algoritmusok keresésére, megismerésére, potenciális informatikai tehetségük fejlesztésére, kibontakoztatására.

A tervezett kutatásom így egy feltáró kutatás, melyben a következő témaköröket vizsgáltam:

- Hány évesen, milyen környezetben találkozott először számítógéppel, informatikai eszközzel?
- Iskolai, vagy egyéb szervezett oktatás keretében tanult-e informatikát?
- Programozást, algoritmizálást tanult-e?
- Hogyan viszonyult, hogyan viszonyul ma a digitális kultúrához?
- Informatikai ismeretei mennyire segítik jelenlegi munkáját?
- Véleménye szerint milyen gondolkodási mód, illetve milyen kompetenciák szükségesek a digitális kultúra megismeréséhez, az infokommunikációs eszközök megfelelő használatához?

A mélyinterjú jellegéből fakadóan a beszélgetések bár a kérdések mentén zajlanak, kötetlenebbek, így az interjúalanyokkal folytatott beszélgetések alkalmával néhány témakört csak az adott területen jártas személyekkel beszéltem át. Ugyanakkor nyitott voltam új témákra, meghallgattam, amit alanyaim el akartak mondani a témával kapcsolatban.

A mélyinterjúkat részben személyesen, részben telefonon keresztül készítettem el, beszélgető partnereimnek anonimitást ígérve, így az elemzés során monogrammal azonosítom őket. A korosztály beazonosítását segítő alanyaim születési évét is leírtam a monogramjuk után zárójelben. A beszélgetésekről hangfelvételt nem készítettem, a beszélgetés során az elhangzott válaszokat egy-egy előkészített dokumentumban a kérdések után begépeltem. Az interjúk készítésének időszaka 2021. szeptember és október hónapja. Az interjúalanyok kor szerinti megoszlása a célcsoport tekintetében közel egyenletes volt, húszas évei végén járóktól az ötvenes éveik közepén járókig vettek részt a kutatásban. A nemenkénti megoszlás aránytalan volt, egyetlen hölgy került be a jelenlegi mintába, de ez az alacsony elemszámmal is magyarázható. Az elkészült interjúk száma ugyanis az idő korlátozott volta miatt jelenleg 7, kutatásomat azonban a későbbiekben folytatom. Képzettség szerint a megkérdezettek között bölcsész, orvos, gyógyszerész és közgazdász szerepelt.

## 4. Esettanulmányok, interjúk elemzése

A következő alfejezetekben összegeztem az egyes interjúkat, amelyek így egy-egy konkrét példát tartalmaznak. Az interjúk során feltárt hasonlóságokat, közös pontokat az összegzés fejezetben ismertetem. Az elemzések közlési sorrendjét a digitális kultúrában való jártasság szerint határoztam meg, a programozásban is jártas személyektől a haladó felhasználókig. A programozásban kevésbé otthonos diákok, megfelelő felhasználói ismerettel – akár csak a saját területükre készült alkalmazások tekintetében – szintén kiemelkedhetnek munkatársaik köréből a középiskolában szerzett kompetenciáikkal.

### 4.1. Orvos végzettségű gyógyszerkereskedő

SG (1967) úgy érzi, a véletlen műve, hogy orvos lett és nem informatikus. Gimnázium első évében kapta az iskola az első számítógépét, és az iskolában szervezett szakkörön megismerkedett a BASIC

programnyelvel. Nagyon megszerette a programozást gimnáziumi évei alatt, versenyen is indult, ahol egy lottóhúzás programért különdíjat kapott. Édesapja munkahelyén a fizetések címletelését előkészítő és a címleteket összegző alkalmazást fejlesztett. A fakultáció választás során a matematika-fizika szakot választotta. Harmadik év szeptemberében az első fakultatív matematika óráról késett a tanár. Osztálytársai várakozás közben arról beszélgettek, hogy az idősebb diákok szerint az iskola legszigorúbb tanárához kerültek, amit riasztó történetek mesélésével magyaráztak. SG elbizonytalanodott, a tanár még nem jelent meg, így megkereste osztályfőnökét és megkérdezte milyen más tantárgyat választhat a fizika mellé a matematika helyett. A biológia tantárgy volt az egyéb lehetőség, a fizika-biológia fakultációval pedig az orvosi egyetemre ment, és ott szerzett diplomát. Eltávolodott a professzionális programozástól, de hobbi szinten továbbra is foglalkozott a számítógépekkel és az informatikával.

Egyetem után egy nemzetközi gyógyszergyártó hazai képviselőjének az értékesítő munkatársa, orvoslátogató lett. Az adminisztrációhoz használt Excel munkalapot kiegészítette ellenőrző függvényekkel, később a kitöltést segítő Visual Basic makrókkal. Ezen fejlesztései minden munkatársának megkönnyítették munkáját, ez is hozzásegítette ahhoz, hogy a ranglétrán gyorsan haladt és napjainkban a hazai képviselő egyik vezetője. Az elmúlt tíz évben több komolyabb szoftverberuházás felelős vezetője volt. Az üzleti oldal ismerete mellett programozási ismeretei segítségével nagyon pontosan tudta specifikálni a feladatot, amivel komoly idő és költségmegtakarítást ért el munkáltatója számára.

Véleménye szerint az informatikai munkához egy strukturált, módszeres gondolkodás szükséges, mert maga a programozás is strukturált és logikus. Ez tetszett meg neki középiskolás korában, ezt szerette meg a programozás tanulás során. És ezt szereti ma is, hogy programjain keresztül irányítani tudja az eseményeket, a program mindig azt végzi el, amit a forráskód utasításaiban meghatároz. Kikapcsolódásként mostanában is készít programokat.

## 4.2. Kutató orvos

SK (1994) a fiatalabb korosztályhoz tartozik, akik már beleszülettek a digitális kultúrába. Gyermekként figyelte édesapja számítógépének képernyőkímélőjét, és nagy lelkesedéssel hallgatta programozó nagypapa történeteit a lyukkártyás rendszerekről. Tíz évesen kapta első számítógépét, amin előbb játszott, később a szövegszerkesztéssel ismerkedett. 12 évesen létrehozta első email fiókját. Az általános iskolában nagyon szerette a Comenius Logo programozást, úgy érezte, valós feladatra tudja használni a matematikai ismereteit. Továbbá a programozás olyan felnőttes dolognak tűnt, komolyabbnak érezte magát ilyenkor, és sokkal szívesebben csinálta, mint az egyszerűbb rajzolás feladatokat. Középiskolában csökkent a motivációja, német nyelvet tanulva nem ismerte fel a programozási nyelvben alkalmazott angol szavak jelentését. A weblapkészítést jól megtanulta, az édesanyja által vezetett baba-mama klubnak ő készítette el a honlapját.

A Semmelweis Egyetemen nem volt komoly informatikai képzés, de lehetősége volt felvenni a Pázmány Péter Katolikus Egyetem programozó kurzusát. Az alapozás során a PlanG nyelven írtak programokat, majd C++-ban. A kurzus része volt a matematikai alapok oktatása, ahol analízist és diszkrét matematikát is tanultak. Megszerette a programozást.

Az egyetem befejezését követően az Egyesült Királyságban lett PhD hallgató, parazitológiával foglalkozik. Kutatásához nélkülözhetetlen a programozás, a statisztikai elemzéseket R programnyelven valósítják meg, egyéb munkákhoz pedig Pythonban írja programjait. Ezen programozási nyelveket már a helyi egyetemen tanulta meg, de ezek megértéséhez, tanulásához nagy segítséget jelentett korábbi programozási tapasztalata.

Véleménye szerint az informatikában a lényeglátás, a gondolatok egyszerű megfogalmazása a fontos, ez segít abban, hogy komplex problémák is egyszerű utasításokkal megvalósíthatók legyenek.

### 4.3. Gyógyszerész

SC (1978) felső tagozatos korában kapott szüleitől egy Commodore 64 számítógépet, amit elsősorban játékprogramok futtatására használt. Az általános iskolában bejárt az informatika szakkörre, és ott ismerkedett a BASIC programozási nyelvvel. Kerület és területszámítási programokat készített. A megyei elit gimnáziumban az 1992-1996 időszakban még nem volt számítástechnika óra, édesapja munkahelyén vett részt újabb programozói tanfolyamon. A Semmelweis Egyetem informatika képzése az operációs rendszer memória kezelésére fókuszált, amit hallgatóként nem érzett hasznosnak. A kollégiumban azonban korábbi ismeretei alapján a diákok számára elérhető 5 számítógép rendszergazdája lett. Egy hallgatótárs kérésére telepített Visual Basic környezetet, amivel ő is ismerkedni kezdett.

Informatikai ismereteinek köszönhetően került be egy egyetemi kutató projektbe, ahol kémiai adatbázist építettek, és a különböző kémiai szerkezeteket mátrixokkal írták le. SC kezdetben „tolmács” szerepet töltött be, ő tudta elmagyarázni a programozóknak, mire is gondolnak a kutatók. A kutatási projekt nem csak a szakdolgozatához, hanem doktori disszertációjához is elegendő forrást biztosított.

Számos Chem-Informatikai alkalmazást ismer, ezeket Visual Basic illetve egyedi script nyelvekben megírt programokkal, scriptekkel konfigurálja, használja kutatásaihoz. Bár elsősorban üzemeltetőnek érzi magát, a konfigurációs és programozási munkát mindig tőle várják munkatársai.

Véleménye szerint a számítógépes gondolkodás türelemre tanít, fel kell ismerni a digitális eszközök korlátait. A programozás tanulás erősítette logikus és analitikus gondolkodását, amely segítségével könnyebben átlát és összerak rendszereket, mint kutató kollégái.

### 4.4. Szótárszerkesztő

MT (1972) a számítógépes játékokkal került a digitális kultúra közelébe. Középiskolában annak idején csak néhány számítógép volt, szakkör keretében ismerkedhettek vele a diákok. Kíváncsi volt a működésére, a maga számára megfogalmazott számítási feladatokat oldott meg az akkor elterjedt BASIC programnyelv segítségével. Sikerélményei voltak a programozásban, tudta, hogy ez a jövő, de az angol nyelv és a nyelvészet közelebb állt hozzá, így a bölcsészkaron tanult tovább. Az Akadémiai Kiadó szótárfelújítási projektjéhez keresett egyetemi hallgatókat diákmunkára, MT jelentkezett. A szerkesztői munka forrása a korábbi szótárak SGML/XML formátumban tárolt változata volt, amit egy dokumentumkezelésre fejlesztett, verziókövetéssel rendelkező alkalmazás segítségével lehetett szerkeszteni. Az első képzés alkalmával kiderült, hogy MT korábbi számítógépes tapasztalatai komoly szakmai előnyt jelentenek. Sokat segített szerkesztő és segédszerkesztő társainak, de a főszerkesztőknek is. Az SGML/XML szerkezet nyelv felől indokolt módosításait már ő tárgyalta le a kiadó munkatársaival. A projekt végén a kiadó felvette alkalmazottnak, lényegesen versenyképesebb bérrel, mint amit akkor bölcsészdiplomával el lehetett érni.

MT véleménye szerint a programozás tanulás során megerősödött gondolkodásában egy strukturáló és szelektáló folyamat. A nagyobb problémák részekre bontása, és részfeladatok megoldása az élet minden területén hasznos. Másfelől egy adott problémakörhöz kapcsolódó társ- és részproblémák közül ki kell tudni választani a feladat szempontjából fontosakat és el kell tudni engedni az elhanyagolhatókat. Később ismerte fel, hogy gondolkodásának ezen összetevőit a programozás tanulásnak köszönheti.

További versenyelőnyt jelentett MT számára a „tolmács” szerep. Az informatikusok nehezen mélyülnek el a lexikográfia tudományában, míg a szótárkészítők kevésbé ismerik az informatika világát. A két szakterület megfelelő kommunikációjához nagy segítség, ha valamilyen módon közelelhetőek az álláspontok. Ebben a tevékenységben tudott MT nagyon komoly segítséget nyújtani a szótárfelújítási munkákban. A szótári DTD (document type definition, az SGML/XML állományokban használható címkékre vonatkozó, reguláris kifejezésekben felírt szabályok) tervezését in-

formatikusok végezték. A szerkesztési munkák során azonban számos új helyzet állt elő, amit a DTD nem engedélyezett. MT lett az a szakértő, aki a legoptimálisabb döntéseket tudta hozni, mely esetben kell módosítani a DTD-t, és mely esetben a szerkesztőnek kell a szócikket hozzáigazítani az érvényes szabályokhoz. Ez utóbbi megoldások a szótár következetességén is sokat javítottak.

#### 4.5. Nyelvtanárból lett hálózattervező

HA (1973) általános iskolás korában kapott egy C16-os számítógépet. Nem csak a játékok érdekelték, megtanulta a BASIC programnyelv alapjait. Középiskolában nem volt még informatika óra, de abban az időben semmi sem érdekelt. Középiskola végén főiskolára jelentkezett angol nyelvtanári szakra.

Angol nyelvtanárként nem érezte úgy, hogy ez az, amivel szívesen foglalkozik. A munkaerőpiac folyamatosan növekedett az informatikai munkaerő iránti igény, úgy gondolta, ezzel érdemes lehet foglalkozni. Családeltartói szerepe miatt nem ment újabb felsőfokú képzésre, hanem az interneten keresztül gyűjtötte az információkat, milyen területeken van szakemberhiány, és mely területekhez vannak jó online képzések. Választása a CISCO hálózati eszközök konfigurálására és üzemeltetésére felkészítő tanfolyamra esett. Az online tanfolyam elvégzését követően jelentkezett hivatalos CISCO vizsgára, amit sikeresen abszolválta, és ezzel megnyílt előtte az informatikai munkaerőpiac. Egy vezető multinacionális telekommunikációs cég vette fel hálózat tervező pozícióba, ami ugyan jelentősen eltért az online tanfolyamon megismertektől, de hamar beletanult. Több éves tapasztalatot követően egy hasonló nagyságú versenytárhoz ment át, szenior hálózattervező munkakörbe.

A hálózattervezésben a rész és az egész kapcsolatának viszonyát, annak elemzését, ez alapján tervezését, megvalósítását szereti legjobban. Az ifjúkori programozás során hasonló gondolkodást kellett alkalmazni.

#### 4.6. Pénzügyi tanácsadó

KR (1969) a főiskolán találkozott először az informatikával, DOS alapú szövegszerkesztő program kezelése volt a tananyag magja. Programozást is tanultak GW Basic nyelven, de az nem kötötte le. A menü vezérelt alkalmazást azonban megtanulta kezelni, és ez már komoly előnyt jelentett előmenetelben. A grafikus operációs rendszereket megelőzően, egér nélkül, a menü vezérelt alkalmazásokat billentyűkombinációkkal kellett kezelni. A kevésbé felhasználóbarát környezet sokakat elbátortalanított, kevesebben tanulták meg ezeket az alkalmazásokat használni. Az első ügyviteli alkalmazáscsomagok pedig ezt a mintát követték, és a grafikus operációs rendszerek elterjedését követően is még sokáig ezek uralták a piacot. Első munkahelyén egyedül ő volt képes kezelni a bevezetett alkalmazást a kezdeti hónapokban. A használat során felismert kényelmetlenségeket jelezte a szoftverfejlesztő cégnek, jó munkakapcsolat alakult ki közöttük. Később a szoftverfejlesztő cég KR-t ajánlta egy új ügyfelének, ahol gazdasági igazgató pozíciót kínáltak számára. Az együttműködés folytatódott, KR az üzleti életből érkező változásokhoz segítette hozzáigazítani az alkalmazást. KR a felhasználóktól kapott negatív visszajelzések megoldására kitűnő ötletekkel állt elő, ami tovább erősítette a kölcsönösen előnyös együttműködését a fejlesztő céggel. Ma saját tanácsadó cége van.

Az informatikai eszközök alkalmazásához kell egyfajta affinitás és kitarás. Jó, ha az embernek van matematikai, logikai gondolkodása és mellette kellően gyakorlatias. Mindemellett kreatív gondolkodásra is szükség van.

#### 4.7. Nyelvész, Bibliafordító

BL (1980) általános iskolás korában játéknak kapott egy Commodore 64 számítógépet. A kazettás magnó, majd később a hajlékony lemez kezelését muszáj volt megtanulni, hogy az adott játékkal játszani tudjon. A géphez kapott könyvek segítségével megismerkedett a BASIC programozási nyelv alapjaival, ami azonban nem kötötte le. Középiskolában csak szakkör keretében volt informatika

képzés, elsősorban a játék lehetőség miatt csatlakozott, de természetesen a szükséges programozási feladatokban is részt vett.

Felsőfokú tanulmányait bölcsészkaron végezte (olasz-hebraisztika szakon), kollégiumi szobatársai informatika szakos hallgatók voltak. Ők mutatták meg neki az internetet, annak lehetőségeit, a kereső oldalak használatát, ami már további bölcsész tanulmányait is jelentősen támogatta.

Bölcsészként elsősorban szövegszerkesztő alkalmazást használ írásaihoz, és böngésző programokat, azok egyedi keresési lehetőségeit kutatásaihoz. Résztvevője egy teljes bibliafordításra dolgozó projektnek. Ebben a munkában különböző nyelvű bibliafordításokat és számos magyarázatot kell minden egyes mondat fordításához átolvasni. Ezek keresése, tárolása, rendszerezése az alap felhasználói kompetenciánál többet igényel. A fiatalkori programozás tanulás azonban megalapozta a számítógépes gondolkodás egy szintjét, amit BL nagyon jól kihasznál munkái során, különösen az egyedi keresési feltételek logikai kapcsolatainak meghatározása során.

Szakfordítóként csak akkor használja az elterjedt fordítást támogató alkalmazásokat, ha a megbízó ragaszkodik hozzá. Ezen alkalmazások nem veszik figyelembe a magyar nyelv sajátosságait, és többet hátráltatnak, mint segítenek. Nyelvészként és gyakorlott számítógép felhasználóként nagyon hasznos tanácsokkal láthatná el a nyelvtechnológiával foglalkozó cégeket, amennyiben lenne fizetőképes kereslet a magyar nyelvi modulok fejlesztésére.

Az informatikus szakmát kívülállóként szemléli, meglátása szerint a számítógépes gondolkodás legfőbb jellemzője az összefüggések felismerése, ami nélkülözhetetlen a problémamegoldáshoz. Ugyanakkor szükséges egy nagyfokú monotonitástűrés is.

#### **4.8. Fiatal kutatók**

Az Eötvös Loránd Tudományegyetem Informatikai Karán indult egy kezdeményezés, hogy az Egyetem különböző karain működő doktori iskolák hallgatói között kutatási együttműködések jöjjenek létre. Az első személyes találkozó alkalmával a megjelent hallgatók röviden ismertették kutatási témájukat, elért eredményeiket és az esetleges interdiszciplináris kapcsolódási pontokat. Biológus, fizikus, irodalmár, nyelvész, közgazdász, pszichológus és szociológus hallgatók beszámolóiban rendre elhangzott, hogy adatokat gyűjtenek, online kérdőíveket készítenek, az eredményeket pedig R környezetben dolgozzák fel. A kutató pálya ma már minden területen igényli nem csak az alkalmazói kompetenciák meglétét, hanem az adatok kezeléséhez és feldolgozásához szükséges adatmodellezési, a megfelelő feldolgozások végrehajtásához pedig az algoritmizálási és programozási képességeket is.

A doktorandusz hallgatók többségének nem okozott nehézséget a kutatáshoz szükséges alkalmazások megismerése, korosztályuknak megfelelően otthonosan mozognak az infokommunikációs eszközök világában. Az R programozás megismerése azonban már nehezebb feladat, amit gyakran a konkrét feladatok elvégzése közben tanulnak meg. A középiskolai digitális kultúra tantárgyban a programozásra, algoritmizálásra bevezetett magasabb órakeret a jövő kutatóinak is hasznos segítség.

### **5. Összegzés**

Az interjúk elemzése során gyűjtöttem a hasonlóságokat interjúalanyaim pályájából. Ezen információk rávilágítanak arra, milyen tevékenységek segítenek megalapozni a digitális kultúrában való eligazodáshoz szükséges számítógépes gondolkodást. A sikeres alapozás során megfigyelhetők a potenciális tehetségek, akikkel külön is érdemes foglalkozni.

Az informatikai eszközökkel a gyermekek többsége először játékként találkozik. A játékhoz való hozzáférés, a játékban való jó eredmény elérése nem csupán a játék kezelésére, hanem az adott eszköz használatának megismerésére, az eszköz operációs rendszerének megismerésére is motiválja a gyerekeket. A gyermekek figyelmének lekötésére egyre több játék áll rendelkezésre, ezek különböző mértékben segítik a digitális kultúrában való jártasság kialakulását. Érdemes azonban ezek közül

kiválogatni néhányat, amelyek módszertani szempontból is hasznosak, és azokat ajánlani pedagógusoknak és szülőknek. A játékok ugyanis kiváló belépők a digitális kultúra világába, ami a gyermekeket motiválja és segítségükkel szó szerint játszva tanulhatnak.

A játékokon keresztül a digitális kultúra világába bevezetett gyerekeket még az általános iskolában hasznos megismertetni a programozás alapjaival. A 80-as években a BASIC programozási nyelv volt az általánosan használt belépő nyelv, napjainkban már könnyebben érthető, kitűnő grafikus lehetőséget tartalmazó programozási környezetek állnak rendelkezésre. Ilyenek például a Logo programozási nyelv különböző implementációi, vagy a blokkprogramozási környezetek, mint például a Scratch. Az utóbbi években ezen eszközök már robotok, kisebb játékok vezérlésére, programozására is használhatók, aminek köszönhetően egyre több diák motivált a programozásra. Természetesen az alapok megismerését követően, további programozási környezetekkel is meg kell ismertetni a diákokat. Az életkori sajátosságok, valamint a diákok egyéb tantárgyakkal kapcsolatos érdeklődésük figyelembevételével ezek lehetnek általános célú, magas szintű programozási nyelvek, vagy adott szakterületek szimulációs környezetének script nyelvei.

Az alkalmazások kezelése a különböző érintőképernyős informatikai eszközökön már általános a kisiskolás gyermekek körében. A számítógépes környezetben a tantervben szereplő irodai alkalmazások kezelése már nem magától értődő. A menürendszerek, az állománykezelés és szerkesztés jellemző parancsai egyrészt általánosak, másrészt tartalmuk és funkciójuk ismerete részét képezi a számítógépes gondolkodásnak. Az általánosan használt irodai alkalmazások mellett a diákok érdeklődési körének, valamint a különböző tantárgyi igényeknek megfelelően további alkalmazások közös felfedezése is szükséges. Az interjúk megerősítették, hogy az általános irodai alkalmazások mellett, az adott szakma egyedi igényeire kifejlesztett alkalmazások használata is fontos. Az alkalmazások többsége nem egyetlen előre meghatározott algoritmus gépies ismétlését várja el a felhasználóktól, így azok kezelése igényli a digitális kultúrában való jártasságot, az alkalmazások funkcióinak önálló megismerésének, felfedezésének képességét.

A válaszok egy része arra is rávilágított, mik hátráltathatják néha a diákokat a digitális kultúra megismerésében. Az alkalmazások tanítása során a diákok többségét zavarja, amikor kizárólag a lépések gépies ismétlését gyakoroltatják az iskolai tanórán. A diákok érdeklődését felkeltő feladatok és lehetőségek kellene a gyerekeknek, hogy felfedezzék az alkalmazásokat, céljukat, használatukat. Természetesen az összetettebb alkalmazások esetén szükséges a nehezebb funkciók bemutatása, de alapvetően a felfedezés öröme kell felkínálni a diákoknak. A rutin lépések begyakoroltatása csak az ismerkedés időszakában alkalmazható.

A felfedező szellemiségű tehetséges diákok nem szeretik, kifejezetten károsnak tartják azokat a téma lezárásokat, amikor a tanár egy ponton azt mondja, a továbbiak túl bonyolultak. Csak szakemberek számára érthető. A kíváncsiságot, a megismerés iránti tanulói szándékot nem szabad megfékezni, inkább terelni, vezetni kell. Nyitva kell hagyni minden kaput az érdeklődőknek, miközben a lépésenkénti finomítás elvét érvényesítve jelezni kell, hogy lépésről lépésre érdemes elmélyülni akár a fejlesztésben, akár a konfigurációs scriptekben, akár egy komolyabb alkalmazás használatában.

Ugyan csak egyetlen válaszoló jelezte, de ezzel is jó, ha számolnak a digitális kultúrát oktató tanárok, hogy vannak olyan diákok, akik nem tanulnak angol nyelvet. Az ő tanuláskuk támogatására fontos elmondani a programnyelvek utasításaiban előforduló angol szavak jelentését.

Az új tanterv érthető okokból több időt szán a programozás oktatásra. Az iskolák és tanárok között ma is sok különbség van, néhány tanár nem mer belevágni a programozás oktatásba. Az ő bátorításukhoz is szükség lenne a jó tapasztalatok és segédanyagok megosztására.

A kutatás megerősítette, hogy az informatikában tehetséges diákokkal akkor is érdemes külön foglalkozni, ha nem informatikai pályára készülnek. A digitális kultúra bár átjárja a hétköznapi élet szinte minden területét épp úgy, mint a tudományt, az informatikai eszközök szakszerű és tudatos használata még nem általános. A felhasználói kultúra fejlesztése mellett az informatikai eszközök



folyamatos tökéletesítésére is szükség van, ezen folyamat támogatásához pedig nagy szükség van olyan szakemberekre, akik szakterületük mellett az algoritmizálás, programozás alapjaival is tisztában vannak. Az ilyen személyek képesek az adott szakmai igények szakszerű megfogalmazására, a specifikációs folyamatban egy „tolmács” szerep betöltésére az alkalmazás fejlesztői és a felhasználói köre között.

## Irodalom

1. Magyar Közlöny: *A Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról szóló 110/2012. (VI. 4.) Korm. rendelet módosításáról*, Budapest (2020)
2. Ambrus, B.: *Az érvényesülés kulcsa a digitális problémamegoldás*. In: Új köznevelés Vol. 76. No. 7. Budapest (2020)  
<https://folyoiratok.oh.gov.hu/uj-kozneveles/az-ervenyesules-kulcsa-a-digitalis-problemamegoldas> (utoljára megtekintve: 2021.10.15.)
3. Varga, J.: *90 évvel ezelőtt született Kemény János, az informatika úttörője*  
<http://www.mszt.hu/web/guest/kemeny-janos-az-informatika-uttoroje> (utoljára megtekintve: 2020.09.25)
4. Csukonyi, Cs. Münnich, Á. Oravecz, Z.: *A demokratikus értékek megjelenése a középiskolások csoportgondolkodásában*. In: Új Pedagógiai Szemle Vol. 53. No. 11. Budapest (2003)
5. Fűzi, B.: *Az oktatás módszerei II.*  
[http://www.banki.hu/~tk/segedanyagok/altalanos-pedagogia/motivacio\\_ea.pdf](http://www.banki.hu/~tk/segedanyagok/altalanos-pedagogia/motivacio_ea.pdf) (utoljára megtekintve: 2021.11.10)



# Programozási alapfogalmak sorrendje a módszeres és funkcionális programozásban

Szabó Zsanett<sup>1</sup>, Visnovitz Márton<sup>2</sup>

<sup>1</sup>szabo.zsanett@inf.elte.hu

<sup>2</sup>visnovitz.marton@inf.elte.hu

ELTE IK

**Absztrakt.** A programozás oktatásában egy adott tanítási módszer, stratégia kiválasztása nagy mértékben meghatározza, hogy a tanulók milyen sorrendben ismerkednek meg a programozás alapfogalmaival. Magyarországon a programozás tanításának egyik leggyakoribb módszere az úgynevezett módszeres (algorithm-first) programozás, de más megközelítések is léteznek, például a funkcionális programozás eszközeivel való tanítás (functional-first). A két módszerben a fogalmak megismerésének sorrendje számos ponton eltér egymástól. Ebben a cikkben a programozás alapfogalmainak megismerési sorrendjét vizsgáljuk és hasonlítjuk össze a módszeres, illetve a funkcionális programozáson alapuló tanítási módszerek esetén.

**Kulcsszavak:** bevezető programozás, programozás oktatása, programozás alapfogalmai, módszeres programozás, funkcionális programozás

## 1. Bevezetés

A programozás oktatása az informatikaoktatás egyik legfontosabb része, szinte egyidős a számítástechnikával [1]. Tanítására többféle módszer alakult ki az idők során. A programozás, illetve programozással való problémamegoldás megfelelő és hatékony tanításához, a megfelelő fogalomstruktúra kialakításához elengedhetetlen a témakör és a témakörhöz tartozó ismeretanyagok, fogalmak és módszerek alapos végiggondolása. Ez elengedhetetlen a témakör tudatos felépítéséhez, a tananyag és az órák megtervezéséhez.

A tudatosan felépített tanmenet és tematikus terv elkészítéséhez szükség van arra, hogy a saját fogalomrendszerünket átlássuk, tudatosítsuk. Tanárként tisztában kell lennünk azzal, hogy a témakörhöz tartozó fogalmak közül melyek az igazán fontosak, azok hogyan kapcsolódnak egymáshoz, hiszen célunk az, hogy a diákjaink fogalomrendszerében is kialakuljanak azok a kapcsolatok, összefüggések, amelyek később megkönnyítik majd az újabb és újabb fogalmak beépülését az addigiak rendszerébe.

A különböző tanítási és programozási módszerek több szempontból jelentősen eltérnek egymástól. Az egyik nagy különbséget épp a programozáshoz kapcsolódó alapfogalmak megismerési, illetve elsajátítási sorrendje jelenti, de nem csak ez a különbség. Vannak olyan programozási alapfogalmak, amelyek csak az egyik, vagy csak a másik módszer alkalmazása esetén kerülnek elő vagy kaphatnak nagyobb hangsúlyt. Emiatt érdemes átgondolni, hogy mik ezek a különbségek, hogy milyen előnyök, illetve esetleges hátrányok jelennek meg az egyes tanítási módszerek alkalmazása esetén.

Ebben a cikkben két módszert, a módszeres és a funkcionális programozáson alapuló tanítást elemezzük és hasonlítjuk össze a programozási alapfogalmak szempontjából. A kétféle módszer bemutatása és a programozásoktatásban való használatuk vizsgálata után először külön-külön mutatjuk majd be a módszeres és a funkcionális programozás alapfogalmait tartalmazó fogalomhálót. Vizsgáljuk, hogy az egyes módszerek esetén mely fogalmak kerülnek, illetve nem kerülnek elő. Ezen

túl a fogalmak megismerési, elsajátítási sorrendjét is elemezzük a kétféle programozástanítási módszer szerint, majd a két fogalomháló különbségeit, hasonlóságait elemezzük.

A különböző alkalmazási lehetőségek, előnyök és hátrányok miatt a tanítási módszerek általában nem tisztán, hanem vegyesen kerülnek alkalmazásra a tanítási folyamat során. Ennek oka éppen az, hogy mindkettőnél az előnyöket szeretnénk kihasználni, a hátrányokat pedig próbáljuk elkerülni. A két módszer vizsgálata és összehasonlítása után kitérünk arra is, hogy milyen lehetőségek vannak a két módszer együttes alkalmazására.

## 2. A módszeres és funkcionális programozás

Programozás tanításáról tulajdonképpen az 1970-es évektől beszélhetünk a mai értelemben. Az áttörést a strukturált programozás néven megjelent módszertan hozta el, melynek lényeges újítása az volt, hogy kevés, de jól definiált algoritmus és adatszerkezet használatára szorítkozott. Az „*oszd meg és uralkodj*” elv alkalmazásával a problémákat részproblémákra osztotta, valamint a programozási nyelvtől elválasztva absztrakt algoritmusokat és adattípusokat vezetett be, ezek alkalmazásával oldotta meg a programozási feladatokat [1,2,3]. Ezt megelőzően a programozás lényege a gépek belső működésének megértése volt, és szinte minden feladat megoldásához új algoritmust kellett kitalálni.

A módszeres programozás, vagyis algoritmusorientált programozási és programozástanítási módszer [1] leírása alapján az algoritmus megtervezését, megértését teszi a programozási folyamat középpontjába. A teljes programozási folyamatot áttekinti a feladat meghatározásától, specifikációjától egészen a dokumentáció elkészítéséig, de a legnagyobb hangsúlyt az algoritmus és az adatstruktúra megtervezése, az algoritmus helyességének belátása kapja. Mind emellett a programozási folyamat további részeiben is megjelennek algoritmusorientált elemek.

Az algoritmusorientált elképzelés egyik alap gondolata, hogy a tervező a végrehajtó szerepébe képzelheti magát, s így az algoritmus helyességéről lehetnek informális elképzelései. Ez egy szekvenciális végrehajtási elképzelésnél nagyon jól építhet egyéni tapasztalatokra, objektumelvű vagy párhuzamos modellekben pedig csoportok működésében, viselkedésében szerzett tapasztalatokra.

Az algoritmus előállításának alap gondolata a szisztematikus felépítés. A Pólya-féle problémamegoldási módszerhez [4] hasonlóan a módszeres programozás a feladatokat általános feladattípusokra, azok megoldássémáira vezeti vissza, az úgynevezett programozási tételekre [5]. A módszeres programozási módszer az *imperatív* programozási elvet követi, melynek két fő jellemzője a parancs és az állapot. Meghatározott parancsokkal változtatja meg a program állapotát, változókkal dolgozik. Az imperatív elv szerint a program készítése során megadjuk a végrehajtandó **lépések egy sorát**, amelyeket el kell végezni ahhoz, hogy a megoldáshoz, a futtatás eredményéhez jussunk.

Mivel a módszeres programozás fókuszában az algoritmus van, így a tanulók elsősorban nem egy konkrét programozási nyelvet, hanem magát az algoritmikus gondolkodásmódot sajátítják el, ismereteik nem lesznek programozási nyelvhez kötöttek [2,6,7,8].

A funkcionális programozás egy, a módszeres programozástól nagyban eltérő paradigma, melynek alapjai matematikai függvényekben, illetve az azokra épülő lambdakalkulusban [9] gyökereznek. Ebben a programozási módszerben a programot egy függvénynek (kifejezésnek) tekintjük, mely függvény kiértékelése tekinthető a program futásának. A program bemenetét a függvény paraméterein keresztül adjuk meg, a program kimenete a kiértékelte függvény által visszaadott érték [10,11].

Típusát tekintve a funkcionális programozás a *deklaratív* programozási módszerek közé tartozik, melyekben a program működésének folyamata helyett a kiszámítás logikájának leírása van a központban [12]. Az ilyen programozásban a programunk nem más, mint definíciók halmaza, mely definíciók alapján a program kimenete megfelelő bemenetek mellett egy kiértékeléssel előállítható. Ennél a módszernél a programozó azt mondja meg a számítógépnek, **hogy mit számítson ki**, ahelyett, hogy megmondaná, **hogyan** tegye.

### 3. A deklaratív és imperatív megközelítés az informatikaoktatásban

Az előzőkből láthattuk, hogy a módszeres, illetve a funkcionális programozási paradigmák között az egyik legfontosabb különbség, hogy míg a módszeres programozásban *imperatív*, addig a funkcionális programozásban *deklaratív* módon készítjük el a programunkat. Ez a két megközelítés számos más helyen is megjelenik az informatikai képzésben. A deklaratív megközelítéssel találkozhatunk például weboldalak készítésekor, adatbáziskezelésnél vagy táblázatkezelési feladatok megoldásakor. A HTML maga is egy deklaratív leírónyelv, de az SQL lekérdezőnyelvben is egy definíciót adunk meg arra, hogy milyen eredményt szeretnénk látni. Az Excel függvények és képletek alkalmazása tulajdonképpen nem más, mint funkcionális programozás. Az imperatív módszerrel találkozhatunk például prezentációkészítés közben, amikor egy adott dián lévő animációkat állítjuk be, de ide tarthat a robotok vezérlése, illetve a hétköznapi algoritmusok alkalmazása is.

A kétféle megközelítés nemcsak a feladatmegoldásban jelenhet meg, hanem már egy feladat megadásának módjában is. Ugyanaz a feladat feladható egymás után végrehajtandó utasítások sorozataként, vagyis utasításkövető, imperatív módon; valamint úgy is, hogy csak az elvárt eredményt határozzuk meg megfelelő definíciókkal, leírásokkal, vagyis deklaratív, mintakövető módon. Az ilyen jellegű utasításkövető vagy mintakövető feladatleírások megjelenhetnek számos témakörben, legyen szó dokumentum-, prezentáció- vagy weboldalkészítésről vagy képszerkesztési feladról (**1. ábra**).

<ol style="list-style-type: none"> <li>1. Rajzolj egy 2,5 cm × 12,5 cm méretű piros (#FF0000) téglalapot!</li> <li>2. Rajzolj egy azonos méretű fehér (#FFFFFF) téglalapot, amely a piros téglalaphoz felső élével illeszkedik!</li> <li>3. Rajzolj egy zöld (#55B24C) téglalapot ugyanolyan méretben, amely a fehér téglalaphoz felső élével illeszkedik!</li> </ol>	<p>Készíts magyar zászlót, mely három egymáshoz illeszkedő, azonos méretű (2,5 cm × 12,5 cm) téglalaphoz áll, a felső téglalap piros, a középső fehér, az alsó zöld színű.</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**1. ábra:** Egy feladat megadása imperatív és deklaratív módon

A tanulóknak mind a kétféle feladatmegadási vagy feladatmegoldási módszert érdemes megismerni, mivel mind az imperatív, mind a deklaratív módszerek megjelennek a hétköznapi életünkben is: legyen szó egy használati, vagy összeszerelési útmutatóról (imperatív) vagy egy felújítási terv elkészítéséről egy lakás felújításához (deklaratív). Mivel ezek a koncepciók a hétköznapi életünkben is erősen jelen vannak, a természetes gondolkodásunk részét képezik, ezért fontos mind a kettő feladatmegoldási módszert megismertetni a tanulóinkkal. Ennek remek eszköze lehet a módszeres és a funkcionális programozás tanítása az iskolában.

### 4. A módszeres és funkcionális programozás megjelenése informatikai tantervekben

A módszeres és a funkcionális programozás is számos tantervben, tanmenetben, módszerben megjelenik, mint a programozásoktatás eszköze. A különböző képzési szinteken ezeknek a paradigmáknak más-más szerepe lehet az oktatásban, ezért különböző képzési szinteken eltérő formákban jelenhet meg a használatuk.

A felsőfokú programozásoktatásban a módszeres programozás tanítása a mai napig meghatározó. A strukturált és algoritmikus programozással kapcsolatos ismeretek gyakorlatilag minden felsőfokú programozásképzésben megjelennek, sőt, más, nem informatikai képzések programjában is

előfordulnak. Gyakori módszer a programozás bevezetéséhez a módszeres programozás, azon belül is a programozási tételek használata.

A funkcionális programozás mint módszer használata az informatikai iparban erősen meghatározó, ezért magától értetődő, hogy a legtöbb felsőfokú informatikusképzés foglalkozik a témával. Habár a funkcionális programozás a legtöbb egyetemi programozásképzésben megjelenik, intézményenként komoly eltérések vannak abban, hogy a képzési tervben hol jelenik meg ez a paradigma. A programozás alapjai megtaníthatók mind módszeres, mind funkcionális programozás segítségével [7,13], sőt az egyes paradigmák közötti átjárhatóság is biztosított [14]. Éppen ezért a döntés, hogy a programozást melyik paradigma segítségével vezetjük be, módszertani megfontolásokon kell alapuljon, melynek összhangban kell állnia az adott képzés céljaival [15,16].

Bizonyos egyetemek (pl. ELTE, University of Cambridge, University of Oxford) már az első félévtől kezdve tanítják a funkcionális paradigmát a módszeres programozás mellett, így a hallgatók (részben) ezen keresztül (is) ismerkednek meg a programozás alapjaival. Arra is találunk példát a világban, hogy a programozási bevezető kizárólagosan funkcionális nyelvek segítségével történik (pl. Imperial College London), a módszeres programozás alapjaival pedig csak ezt követően ismerkednek meg a hallgatók. Ezzel szemben bizonyos egyetemeken (pl. BME) a funkcionális programozás csak későbbi félévekben jelenik meg, a programozás bevezetésére kizárólagosan a módszeres programozást használják.

Természetesen a kétféle paradigma nem csak a felsőoktatásban kerül elő. A 2020-ban bevezetett Nemzeti Alaptantervben (továbbiakban NAT) [17] leginkább a módszeres programozás alapjai jelennek meg. Olyan kulcsszavakat találunk a dokumentumban, mint algoritmizálás, illetve ahhoz kapcsolódó fogalmak, például „*vezérlési szerkezet*”. A funkcionális programozás alapjait lényegesen nehezebb megtalálni a NAT-ban. A NAT programozással kapcsolatos témaköreiben nem kerül közvetlenül említésre a funkcionális programozás, de a függvények használata megjelenik a matematika tantárgynál, illetve a digitális kultúra tantárgyon belül a táblázatkezelés témakörben is. Ezek alapján mondhatnánk, hogy a közoktatásban a funkcionális programozás mint olyan nem jelenik meg, de a függvények használata a táblázatkezelés témakörben alapvetően nem sokban különbözik a klasszikus funkcionális programozástól. Ezt mi sem mutatja jobban, mint hogy léteznek módszerek, melyek kifejezetten a táblázatkezelésre alapozva, funkcionális programozáson keresztül vezetik be a tanulókat a programozás világába [18], majd ezt követően az így szerzett tudást használják fel az algoritmikus gondolkodás fejlesztésére [19].

Az érettségi vizsgán az aktuális, 2012-es NAT-on [20] alapuló követelményrendszer [21] szerint programozási feladat csak emelt szinten van, de egyszerű függvények használata a táblázatkezelés és adatbáziskezelés feladatnál is elvárás. Habár emelt szinten alapvetően a módszeres programozás szerint vannak meghatározva a követelmények (pl. ismerje a mondatszerű algoritmus-leíró eszközt; tudja használni az elemi programozási tételeket; legyen képes egy mondatszerű leírással készült algoritmust a használt programozási nyelvben kódolni), a programozási feladat egy-két dolgot leszámítva (például fájlból való beolvasás és kiírás) tisztán funkcionális programozással is megoldható.

A 2020-as NAT-on alapuló, 2024. január 1-től érvényes érettségi követelmények [22] alapján már középszinten is elvárás az algoritmizálás és a programozás, de az eljárások és függvények ismerete csak emelt szinten elvárás, középszinten nem. (Korábban emelt szinten sem szerepelt a követelmények között.) A követelmények a módszeres és a funkcionális programozás használhatóságát illetően nem változtak.

Az érettségi feladatsorok programozás feladatai fájlból olvasással kezdődnek, majd a beolvasott adatok feldolgozása után az adatokra vonatkozó kérdések következnek, melyek általában egy-egy programozási tétellel vagy azok összeépítésével oldhatók meg. A 2020-as NAT-on alapuló, 2024-től bevezetésre kerülő vizsgakövetelmények szerint készített mintafeladat alapján az új feladatsorokban a korábbiaknál összetettebb kérdésekre lehet majd számítani emelt szinten, de továbbra is igaz, hogy

a feladatok megvalósíthatók a programozási tételek alkalmazásával és azok összeépítésével. A programozási tételek módszeres és funkcionális módon is megvalósíthatók [14], így a feladatok megoldása mind a két paradigma használatával lehetséges.

A digitális kultúra tantárgy 2024-től bevezetésre kerülő érettségikövetelmények szerint készített középszintű mintafeladatsorának [23] programozási feladatára viszont már nem mondható el ugyanaz. A mintafeladatban a program készítésének lépései imperatívan, lépésről lépésre vannak meghatározva. Konkrét feladatként szerepel bizonyos (előre meghatározott nevű) változók létrehozása, értékadások, valamint ciklus és elágazás létrehozása is. Emellett szerepel véletlen számok generálása is, amihez viszont mindenképp szükség van a véletlen számot generáló függvény alkalmazására. A feladatmegoldás során végrehajtandó konkrét utasítások miatt a középszintű mintafeladat alapvetően nem oldható meg funkcionálisan.

## 5. Programozási fogalmak sorrendje és egymásra épülése

A számunkra, illetve a tanított csoport számára megfelelő programozástanítási módszer kiválasztásához fontos végiggondolni egyebek mellett azt is, hogy a különböző módszerek választása esetén mely programozási alapfogalmakat sajátíthatják el a tanulók, hogy mi a programozás tanításának célja az adott csoport esetén, hogy mennyi idő áll rendelkezésre a témakör tanításához.

A fogalmak összegyűjtéskor szembesülnünk kell azzal, hogy az algoritmizálás, programozás témakör rengeteg fogalmat használ, a fogalmak közötti kapcsolatok viszont sok esetben nem egyértelműek. Nem, vagy csak nagyon nehezen határozható meg a fogalmak tanításának „helyes” sorrendje, és egy ilyen „helyes” sorrend pedig számos tényezőtől függ. Az egyik tényező az, hogy milyen környezetben, milyen programozási nyelv tanulása közben kerülnek elő ezek a fogalmak.

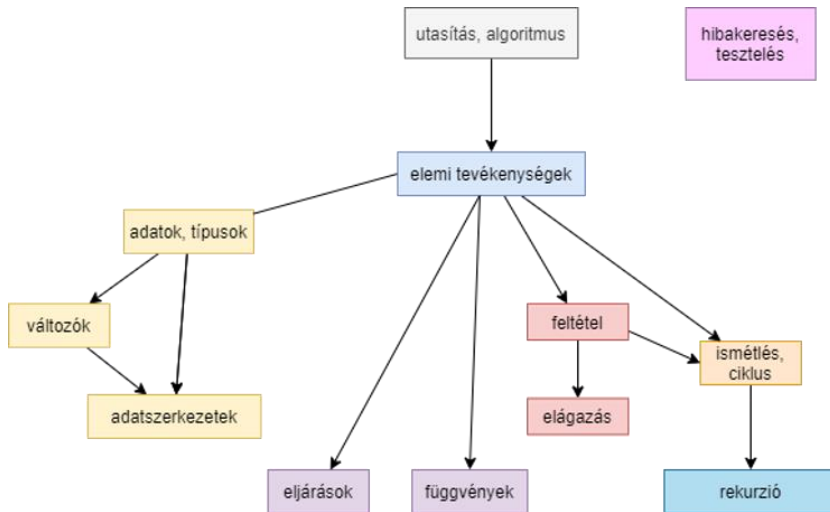
Az algoritmizálás, programozás témakörhöz tartozó alapfogalmak egyfajta általános rendszerezését egy korábbi kutatás [5] mutatja be. A teljes fogalomstruktúra meglehetősen komplex, ezért ebben a cikkben nem a konkrét fogalmak, hanem nagyobb fogalomcsoportok kapcsolatát és tanítási sorrendjét vizsgáljuk majd a módszeres és a funkcionális programozás esetében.

Minden fogalomcsoporthoz több fogalom tartozik, melyek kapcsolatai, tanítási sorrendjük szintén fontos, de a módszeres és funkcionális programozás általunk szemléltetni kívánt különbségei már a fogalomcsoportok szintjén is jól láthatók. Alaposabb vizsgálat esetén további különbségek figyelhetők meg a kétféle módszer fogalomstruktúrájában az egyes fogalomcsoportokon belül.

A fogalomhálókbán a fogalomcsoportok közötti kapcsolatok mellett azok tanítási sorrendjét is igyekeztünk szemléltetni. Minél lejjebb került egy fogalom a kapcsolatrendszereket bemutató ábrán, várhatóan annál később kerül elő a tanulási, tanítási folyamatban. A fogalomcsoportok tanítási sorrendje nem teljesen rögzített. Néhány esetben, a fogalmak közötti kapcsolatból adódó sorrend kötött, de ezek mellett több fogalomcsoport tanítási sorrendje módosítható a tanított csoport igényeinek és a tanítás céljának megfelelően.

## Módszeres programozás

Módszeres programozás esetén a fogalomcsoportokba szervezett fogalomháló a **2. ábra** szerint alakul.



**2. ábra:** A programozás fogalomcsoportjainak kapcsolatai módszeres programozásban

A módszeres programozás fogalomhálójában, ahogyan ezt a módszer másik neve, az algoritmusorientált programozás is mutatja, az elsőként megismerésre kerülő fogalomcsoport az *utasításhoz* és magához az *algoritmushoz* köthető. Az utasítás és algoritmus fogalmaihoz elsőként az *elemi tevékenységek* (pl.: beolvasás, kírás) kapcsolódnak. A legtöbb nyelvben és fejlesztőkörnyezetben egy új „üres” program automatikusan tartalmaz egy kírást. Az elemi tevékenységekhez szorosan kötődnek az *adatok, típusok*, azokhoz pedig a különböző *változók* és *adatszerkezetek*.

Az elemi tevékenységekhez kapcsolódó másik fogalom, ami általában nagyon hamar előkerül, az a *feltétel*, hiszen feltétel szükséges az *elágazások* és *ciklusok* használatához is. A fogalomstruktúra egyik legfontosabb és a fogalomcsoporthoz kapcsolódó fogalmak számát tekintve egyik legnagyobb részét képezik a *feltételek*hez, *elágazás*hoz és a *ciklusok*hoz kapcsolódó fogalmak. Ezek tisztázása, illetve megértése különösen fontos és elengedhetetlen része a témakör tanulásának, illetve tanításának, hiszen az algoritmusok és programok ezekből a vezérlési szerkezetekből építkeznek.

Az *eljárások* és *függvények* a módszeres programozási módszer esetén általában csak később, az alapvető vezérlési szerkezetek használatának begyakorlása után kerülnek elő, ha egyáltalán előkerülnek. Ezek csupán a kódszervezés eszközei, az elkészített program hatékonyabbá, átláthatóbbá tételét segítik, de minden megoldható nélkülük. Elegendő csak a főprogramhoz tartozó függvényt/eljárást használnunk, amiről a programozással még épp csak ismerkedő tanulóknak nem is kell feltétlenül tudniuk, hogy az egy függvény/eljárás. Kezdetben számukra csak annyi a fontos, hogy minden utasítást, amit végre szeretnének hajtani, azt a főprogramon (pl. main függvényen) belülről kell írniuk. Ha kevés a témakör tanítására fordítható idő, akkor az is előfordul, hogy nem is találkoznak olyan feladatokkal a tanulók, ahol függvényeket, eljárásokat használnak, vagy csak egy-egy példán keresztül kerül bemutatásra, mint fejlesztési lehetőség.

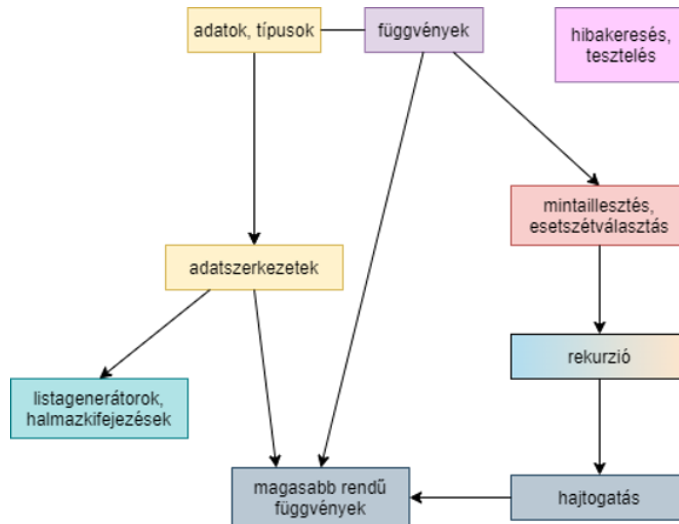
A *rekurzió*, a függvényekhez és eljárásokhoz hasonlóan középiskolában érdekességként kerül csak elő a módszeres programozásoktatásban. Ez nem szerves és elengedhetetlen része a témakör tanításának, az alapvető programozási ismeretek elsajátításának.



A programok futtatásakor és az elkészített algoritmusok ellenőrzésekor *hibakereséssel* és *teszteléssel* is foglalkoznunk kell. Ezek végigkísérik a teljes programozástanulási folyamatot, de közvetlenül nem kapcsolódnak a többi programozási alapfogalomhoz. Az ide tartozó fogalmak és hibakeresési módszerek nagy mértékben függenek a programozási környezettől, illetve a témakör tanításának mélységétől, de minimális szintű hibakezelésre a választott környezettől függetlenül szükség van már a legegyszerűbb programok elkészítése során is (pl. kódolási hibák javítása).

### Funkcionális programozás

A funkcionális programozás legfontosabb fogalma a *függvény*, a paradigma a nevét is innen kapja (angolul: function). A funkcionális programozásban majdnem minden további koncepció közvetve vagy közvetlenül ehhez a központi fogalomhoz köthető, így értelemszerűen ez áll a fogalmak rendszerének központjában (**Hiba! A hivatkozási forrás nem található.**), illetve a sorrendiséget tekintve nagyon korán szerepel. A programozási függvényfogalom bevezetéséhez felhasználhatók korábban megszerzett előismeretek, mint például a matematikai függvényfogalom vagy a táblázatkezelésben használt függvények ismerete.



3. ábra: A programozás fogalomcsoportjainak kapcsolatai funkcionális programozásban

A másik fontos alapfogalom a funkcionális programozásban a típus, mely szintén szorosan köthető a függvényfogalomhoz. A klasszikus, tisztán funkcionális nyelvekben minden típus valójában a függvénynek egy altípusa, de nem feltétlenül szükséges a típusfogalmat ilyen megközelítésben bevezetni. Az egyszerű típusok (logikai, szám, karakter, stb.) önmagukban is bevezethetők és később köthetők a függvényekhez a bemeneti paraméterek típusa és a visszatérési típus révén. Már ezen a ponton komoly különbségek lehetnek a fogalmak bevezetésében a programozási nyelv választásától függően. Egy tisztán funkcionális, erősen típusos nyelv (pl. Haskell) esetén nem tudunk saját függvényt definiálni anélkül, hogy típusokat adnánk meg a függvény szignatúrájában, míg egy dinamikus vagy gyengén típusos nyelv (pl. JavaScript) esetén lehetőségünk van erre.

A további fogalmi építkezést szintén erősen befolyásolja a nyelvválasztás. Számos nyelv rendelkezik olyan szintaktikai megoldásokkal, mellyel könnyen bevezethető a *mintaillesztés* és az *esetszétválasztás* fogalma, melyek a módszeres programozásban az elágazások analógiái. Ezen fogalmak ismeretében lehetőségünk nyílik megismerni a rekurzió fogalmát, melynek – azon túl, hogy egy általánosságban is fontos feladatmegoldási módszer – a funkcionális programozásban, a komplex adatszerke-

zetek feldolgozásában is jelentős szerepe van. Komplex adatszerkezetek (pl. lista) rekurzióra épülő feldolgozását jellemzően a *hajtogatás* módszerének segítségével tudunk megtenni.

A mintaillesztés és a rekurzió mellett egy másik vonalon, a *típus* és az ehhez szorosan kapcsoló *adat* fogalmainak bevezetése után lehetőség van továbblépni a haladóbb, *összetettebb adatszerkezetekre*, mint például a rendezett n-es, a lista vagy éppen a rekord. Ezek az összetett típusok már lehetőséget adnak az adatmodellezéssel kapcsolatos ismeretek elsajátítására, elmélyítésére, és utat nyitnak a funkcionális programozás bizonyos haladóbb lehetőségei felé. Ezen haladó lehetőségek közé tartoznak például bizonyos nyelvekben (pl. Clean, Haskell) a *listagenerátorok*, illetve *halmazkifejezések*, melyekkel deklaratív módon tudunk összetett adatszerkezet-példányokat definiálni.

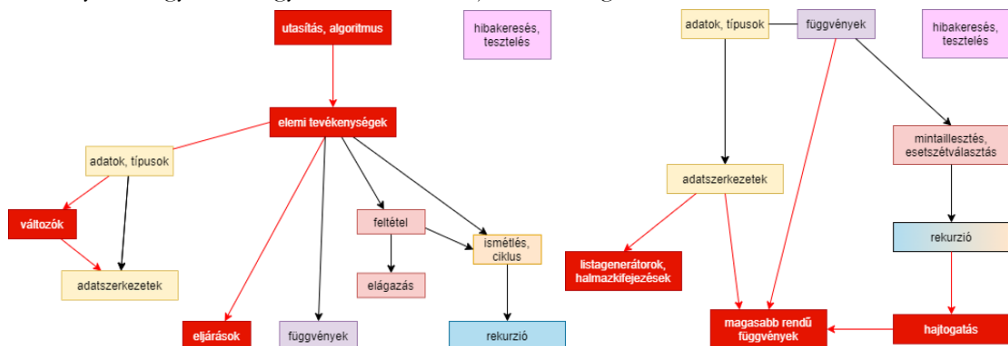
Egy másik irány, amerre tovább haladhatunk az összetett adatszerkezetektől, az a magasabbrendű függvények világa. Ezen a ponton már szükségessé válik a függvény mint elsőrendű típus fogalmának bevezetése, ha ez korábban nem történt meg. A magasabb rendű függvények igencsak hatékony eszközök a feladatmegoldásban. Segítségükkel lehetőségünk nyílik komplex adatfeldolgozási feladatok megoldására oly módon, hogy a feldolgozás logikája helyett (pl. hogyan iterálunk végig egy listán) a konkrét megoldandó feladat specifikus részeire tudunk koncentrálni (pl. egy kiválogatás esetén mi a kiválogatás feltétele).

A magasabb-rendű függvényekig a hajtogatáson, és a magasabb-rendű függvények implementálásán keresztül vezető úton is eljuthatunk. Szinte minden, a funkcionális programozást támogató nyelvben előre definiálva megjelennek ezek a magasabb-rendű függvények, melyek közül a leggyakoribbak (reduce/fold, map, filter, find, any/some, all/every stb.) megfelelnek az algoritmikus programozásban tanított programozási tételeknek [14].

Természetesen a módszeres programozáshoz hasonlóan itt is végigkísérik a fogalmi hálót a teszteléssel és hibakereséssel kapcsolatos ismeretek. Habár ezek módja sok helyen eltér a módszeres programozásban alkalmazott módszerektől, a kapcsolódó fogalmak (pl. szintaktikus, szemantikus hiba) hasonlóak.

## 6. Fogalmak a módszeres és funkcionális programozásban

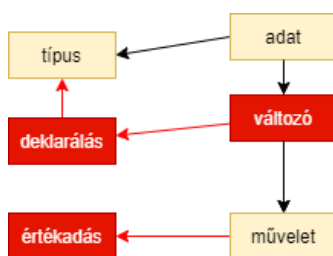
A módszeres és a funkcionális programozáshoz tartozó fogalomhálókat ismertetése alapján is könnyen belátható, hogy jelentős különbségek vannak a kétféle módszer alkalmazása esetén előkerülő fogalmak között, illetve a fogalmak tanítási sorrendjében. A 4. ábra a két fogalomháló összehasonlítását segíti. Egymás mellett látható a két fogalomháló és pirossal jelöltük azokat a fogalomcsoportokat, amelyek az egyikben vagy a másikban nem jelennek meg.



4. ábra: A módszeres és a funkcionális programozási módszerek fogalomhálóinak összehasonlítása fogalomcsoportok szintjén

A funkcionális programozás fogalomhálójában nem kerül elő az utasítás és algoritmus, az elemi tevékenységek, a változók és az eljárások, a módszeres programozás fogalomhálójából pedig a magasabb rendű függvények, listagenerátorok és halmazkifejezések, valamint a hajtogatás hiányzik a funkcionális programozás fogalomhálójához képest.

A két fogalomháló között nemcsak fogalomcsoportok szintjén, hanem az egyes fogalomcsoportokon belül is vannak különbségek. Ebből következően a fogalomháló azon részei sem teljesen egyeznek meg, amelyeket a fogalomcsoportok struktúrája alapján azonosnak gondolunk. Az 5. ábra segítségével példaként megmutatjuk, hogy az *adatok*, *típusok* fogalomcsoport hogy néz ki fogalmak szintjén. Módszeres programozás esetén mind a hat fogalom (*adat*, *típus*, *változó*, *deklarálás*, *művelet*, *értékkadás*) megjelenik, de funkcionális programozás esetén ezek közül csak három, az *adat*, *típus* és *művelet* kerül elő, mert a klasszikus, tisztán funkcionális programozásban nincsenek változók, így azokat deklarálni sem lehetséges, illetve értéket sem tudunk adni nekik.



5. ábra: Az *adatok*, *típusok* fogalomcsoportban megjelenő különbségek

## 7. Lehetőségek a paradigmák kombinálására

Ahogy az előzőkből láthattuk, a sorrendiségen túl a megismert tartalmak terén is komoly különbségek vannak attól függően, hogy a programozás bevezetésére módszeres vagy funkcionális programozást választunk. Ezek alapján jogosan merül fel a kérdés, hogy mit tehetünk, ha mind a két paradigma fogalmaival szeretnénk megismertetni a tanulókat.

Kézenfekvő megoldás lehet a paradigmák egymás utáni bemutatása. Számos helyen találunk példát erre a megközelítésre, legyen szó egyetemi képzésről vagy iskolai módszertanról. Ha ezt a módszert választjuk, lehetőségünk van kihasználni a fogalmi hálók közötti átfedéseket, ezeket kapcsolódási vagy akár kiindulási pontként használva mikor bevezetjük a diákokat a második paradigma világába. Ennek a megközelítésnek egy lehetséges veszélye, hogy az elsőnek tanult paradigma módszerei, gondolkodásmódja annyira meghatározóvá válik a tanuló problémamegoldásában, hogy a másodikként tanult paradigma használata közben is ezeket a betanult módszereket akarja majd használni, emiatt pedig nehezebbé válik számára a második paradigma elsajátítása.

Egy másik lehetséges megközelítés – amennyiben lehetőség van rá – a módszeres és a funkcionális programozás párhuzamos tanítása (pl. azonos félévben levő egyetemi kurzusok esetén). Ez a megoldás segíthet abban, hogy egyik paradigmát se tekintse a tanuló „elsőlegesnek” csupán azért, mert azzal ismerkedett meg először. Ha ezt a megoldást választjuk, akkor ügyelni kell arra, hogy a kétféle programozási paradigma tanítása megfelelően elkülönüljön egymástól, ne keveredjenek a különálló részek, de fontos a kapcsolódási pontok bemutatása is. Ezzel a módszerrel, amennyiben a két irányzat tanítása formálisan is elkülönül (pl. külön kurzus formájában), jellemzően két különböző programozási nyelven keresztül történik a paradigmák bemutatása. Mind az algoritmikus, mind a funkcionális programozáshoz van lehetőségünk olyan nyelvet választani, mely kiemeli az adott paradigma sajátosságait, amin keresztül annak jellegzetességei könnyen bemutathatók.

A módszeres és a funkcionális programozás kombinálására egy harmadik lehetőség az, hogy olyan programozási nyelvet válasszunk, ami támogatja a feladatmegoldást mindkét módszerrel (pl. Python, JavaScript). Egy ilyen multiparadigmás nyelv használatával nem csak arra nyílik lehetőségünk, hogy egy eszközkészlettel mutassuk be külön-külön a két paradigmát, hanem arra is, hogy a kettőt kombináljuk. Ezáltal lehetőségünk nyílik annak hangsúlyozására, hogy az, hogy többféle programozási paradigma létezik, az nem jelenti azt, hogy egy feladat megoldásában kizárólagosan az egyik vagy a másik mellett kell dönteni. Az ilyesfajta kombinálásra kiváló lehetőség lehet például a magasabb rendű függvények használata, melyeknél a paraméterként átadott függvényt implementálhatjuk akár algoritmikus módon is, vagyis magas szinten funkcionális programozást használhatunk, míg alacsony szinten módszereset. Egy ilyen megközelítés esetén a kétféle paradigmához tartozó fogalmi hálók összefonódnak, a kapcsolódási pontok mentén egy nagyobb, komplexebb fogalmi rendszert kapunk, melyben megtalálhatók a módszeres és a funkcionális programozás fogalmi is. Természetesen itt is nagyon fontos a programozási nyelv kiválasztása, mivel az meghatározza, hogy milyen lehetőségek állnak rendelkezésünkre a feladatmegoldásban. Vannak olyan, kifejezetten a funkcionális programozáshoz köthető megoldások (pl. mintaillesztés, listagenerátorok) melyek a multiparadigmás nyelvekben ritkán fordulnak elő. Értelemszerűen, ha a választott nyelv ezeket nem támogatja, akkor ezek bemutatására, tanítására sincs lehetőség ugyanabban a környezetben, amelyben a többi alapfogalmat bemutatjuk.

## 8. Konklúzió

Összegzésként elmondható, hogy a módszeres és funkcionális programozási módszer egyaránt fontos szerepet játszik a programozás tanításában. Mindkét módszerre láthatunk példát az oktatás különböző szintjein. Van, ahol csak az egyik vagy csak a másik kerül elő, de az egyetemi képzés során a tanulók jellemzően mindkét módszerrel megismerkednek.

A kétféle módszer nem csak a programozásoktatásban, hanem a hétköznapi életünkben is jelen van, hiszen gyakran találkozhatunk a módszeres programozásra jellemző imperatív feladatmegadással, ahol lépésről lépésre meg van határozva, hogy mit kell tennünk (pl.: automaták használata, szerelési útmutatók). Emellett pedig az is gyakori, hogy a funkcionális programozásra jellemző deklaratív feladatmegadással találkozunk, ahol csak az elvárt eredmény van meghatározva (pl. legyen tiszta a lakás – mit értünk tiszta lakás alatt).

A programozás tanításához választott módszer (módszeres vagy funkcionális) meghatározza, hogy mely fogalmakkal ismerkednek meg a tanulók és a fogalmak elsajátításának sorrendjét is befolyásolja. Jelentős különbségek vannak a két módszer alkalmazása esetén felépülő fogalomhálóban már fogalomcsoportok szintjén is, de az egyes fogalomcsoportokon belül, fogalmak szintjén is vannak további eltérések.

Az egyetemi képzések felépítésekor fontos módszertani kérdés, hogy milyen sorrendben jelenjen meg a módszeres és a funkcionális paradigma, többféle megoldásra mutattunk példát. Középiskolában az érettségi által meghatározott kimeneti követelmények miatt alapvetően a módszeres programozásoktatás jellemző, de van ellenpélda is, illetve olyan is, hogy már középiskolában találkoznak mindkét módszerrel a tanulók. Középiskolák esetén a tanárok, iskolai munkaközösség felelőssége, hogy a tanított csoportnak megfelelő módszert válasszák.

A középiskolai tanárok, illetve azok számára is nagy segítséget jelenthet, ha kombinálni tudják a módszeres és funkcionális programozás módszereit, akik mindkettő fogalmaival és előnyeivel meg szeretnék ismertetni a tanítványaikat. Erre jó lehetőséget biztosítanak a multiparadigmás környezetek. Ilyen környezet választása esetén a két módszer fogalomhálójának összekapcsolásával szélesebb

körű ismeretek adhatók át, de megmarad annak lehetősége is, hogy a tanított csoportnak megfelelő mélységben és részletességgel tárgyaljuk a programozás témakör bizonyos részeit, fogalmait.

## Irodalom

- [1] Szlávi, P., Zsakó, L.: *Az informatika oktatása*. (2011)
- [2] Dijkstra, E. W.: *A discipline of programming*. Prentice-Hall, (1976)
- [3] Dahl, O.-J., Dijkstra, E. W., Hoare, C. A. R.: *Structured Programming*. Academic Press: New York, (1972)
- [4] Pólya, G.: *A gondolkodás iskolája*. Akkord Kiadó, (2000)
- [5] Szabó, Z.: *Problem Solving and Interrelation of Concepts in Teaching Algorithmic Thinking and Programming*, In: Proceedings of the 11th International Conference on Applied Informatics (ICAI 2020), pp. 318–327, (2020), [Online]  
<http://ceur-ws.org/Vol-2650/paper33.pdf>
- [6] Szlávi, P., Zsakó, L.: *Módszeres programozás: Programozási bevezető*. In: Mikrológia vol. 18. ELTE TTK Általános Számítástudományi Tanszék, Budapest, (1994)
- [7] Szlávi, P., Zsakó, L.: *Módszeres programozás: Programozási tételek*. In: Mikrológia vol. 19. ELTE Informatikai Kar, (2008)
- [8] Fóthi, Á.: *Bevezetés a programozásboz*. ELTE Eötvös Kiadó: Budapest, (2005)
- [9] Church, A.: *An unsolvable problem of elementary number theory*, In: American Journal of Mathematics, vol. 58, pp. 345–363, (1936), DOI: 10.2307/2268571
- [10] Horváth, Z.: *A funkcionális programozás nyelvi elemei*. In: Programozási nyelvek (szerk. Nyékiné Gaizler, J.). Kiskapu kiadó, (2003), p. 56
- [11] Barendregt, H., Barendsen, E.: *Introduction to lambda calculus*, In: Nieuw archief voor wisenkunde, (2000)
- [12] Lloyd, J. W.: *Practical Advantages of Declarative Programming*, (1994).
- [13] Abelson, H., Sussman, J., Sussman, J.: *Structure and Interpretation of Computer Programs*. MIT Press, (1984)
- [14] Visnovitz, M.: *Classical Programming Topics with Functional Programming*, In: Central-European Journal of New Technologies in Research, Education and Practice, (2020), DOI: 10.36427/cejntrep.2.2.965
- [15] Chakravarty, M. M. T., Keller, G.: *Educational pearl: The risks and benefits of teaching purely functional programming in first year*, In: Journal of Functional Programming, (2004), DOI: 10.1017/S0956796803004805
- [16] Joosten, S., van den Berg, K., van der Hoeven, G.: *Teaching functional programming to first-year students*, In: Journal of Functional Programming, vol. 3, no. 1, pp. 49–65, (1993), DOI: 10.1017/S0956796800000599
- [17] *Nemzeti Alaptanterv 2020*. (2020)
- [18] Csernoch, M., Bíró, P.: *Sprego Programming*, In: Spreadsheets in Education, vol. 8, no. 1, (2015)
- [19] Csernoch, M., Bíró, P., Máth, J.: *Developing Computational Thinking Skills with Algorithm-Driven Spreadsheets*, In: IEEE Access, vol. 9, (2021), DOI: 10.1109/ACCESS.2021.3126757
- [20] *Nemzeti Alaptanterv 2012*. (2012)
- [21] Oktatási Hivatal: *Informatika részletes érettségi vizsgakövetelmények (2017. május-júniusi vizsgaidőszakról)*

[https://www.oktatas.hu/pub\\_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2017/informatika\\_vk.pdf](https://www.oktatas.hu/pub_bin/dload/kozoktatas/erettsegi/vizsgakövetelmények2017/informatika_vk.pdf)

- [22] Oktatási Hivatal: *Digitális kultúra részletes érettségi vizsgakövetelmények (2022. január 1-től)*. (2021)
- [23] Oktatási Hivatal: *Digitális kultúra középszintű írásbeli érettségi mintafeladatok a 2024. január 1-től bevezetésre kerülő vizsgakövetelmények szerint*.