

Játékkal tanulni, tanulással játszani: tapasztalatok az első középiskolai élményinformatikai táborunkról

Dienes-Gulácsi Nikolett¹, Gulácsi Ádám²

{¹dienes.nikolett, ²gulacsi.adam}@diosgyorigimnazium.hu
Diósgyőri Gimnázium, Miskolc

Absztrakt. Pályakezdő pedagógusként kifejezetten hamar szembesülni lehet az oktatás egyik nehézségével napjainkban. Nevezetesen, hogy az internet világában felnövekvő generáció motiválása az osztályteremben korántsem egy egyszerű feladat.

Egy olyan egyhetes tábort szerveztünk iskolánk tanulóinak, ahol élményekben gazdag, számukra eddig teljesen ismeretlen közegben fejleszthették informatikai ismereteiket. A résztvevők videójáték fejlesztés, játék tervezés (game design), unplugged feladatok, 3D modellezés és nyomtatás, drón API programozás és multimédiás tartalomgyártás témájú foglalkozásokon szerezhettek új tapasztalatokat.

A tábor célja elsősorban a diákok motiválása volt. Fontos rávilágítani arra, hogy ha valaki jó informatikai alapokkal rendelkezik, a tudomány és az élet más-más területein is hasznosíthatja azt. A digitális kultúra órákon fejlesztett számítógépes gondolkodásukat, algoritmizáló készségüket kiválóan alkalmazták a látványosabb, élménydúsabb témakörökben is a táborban.

Kulcsszavak: játékfejlesztés, játéktervezés, nyári tábor, programozás, algoritmikus gondolkodás

1. Motiváció

A diákok az informatika órákon gyakran motiválatlanok, ami jelentős hatással van a teljesítményükre is [14]. Sőt, a motiváció hiánya olyan tanulóknál is megjelenik, akik egyébként tehetségesek és érdeklődnek az informatika tudománya iránt. Az egyik fő probléma a teljesítmény alapú célorientált hozzáállás. Az ilyen beállítottságú diákok fő motivációját a jeles osztályzatok megszerzése, valamint a minél sikeresebb érettségi vizsga teljesítése adja [14]. Az effajta motiváció negatív hatással van a kognitív teljesítményükre és gyakran a Kahneman által körülírt lassú gondolkodást alkalmazzák olyan helyzetekben is, amikor gyors gondolkodás lenne indokolt [4]. A számítógépes problémamegoldás kontextusában ez azt jelenti, hogy a feladatok megoldása során felületi megközelítésű metakognitív módszereket alkalmaznak [16].

Szintén probléma a középiskolás tanulók esetében, hogy nagyon eltérő tudással érkeznek a gimnáziumi osztályokba. Az iskolákból hozott osztályzataik és a saját bevállalásuk szerinti tudásszintjük sokszor nem tükrözi a valóságot [23]. Ez utóbbi a felsőoktatásban is megjelenik [17].

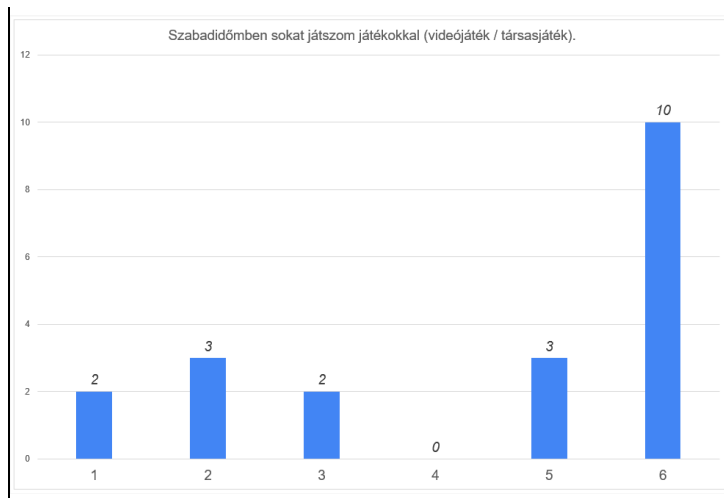
Érdekes módon a 2020-ban bevezetett új NAT már a 3–4. évfolyamra vonatkozóan olyan átfogó célokat fogalmaz meg, amelyek a gimnáziumi évek alatt is relevánsak:

- „1. elmélyülten dolgozik digitális környezetben, önellenőrzést végez;*
- 2. megvizsgálja és értékeli az általa vagy társai által alkalmazott, létrehozott, megvalósított eljárásokat;*
- 3. társaival együttműködve online és offline környezetben egyaránt megold különböző feladatokat, ötleteit, véleményét megfogalmazza, részt vesz a közös álláspont kialakításában;*
- 4. kiválasztja az általa ismert informatikai eszközök és alkalmazások közül azokat, melyek az adott probléma megoldásához szükségesek;*

5. eredményétől függően módosítja a problémamegoldás folyamatában az adott, egyszerű tevékenységsorokat.” [13/141. o.]

A célunk egy olyan tábor szervezése volt, ami többek között a NAT által (is) megfogalmazott célok elérésében segíti a diákokat. Ezen felül a másik fő szempont az informatika tudományában tehetséges és érdeklődő tanulók motiválása. A jártasság alapú célorientált hozzáállásra fókuszáltunk, azaz az érettségi és a jeles osztályzatok helyett olyan célokat tűztünk ki a diákoknak, mint például „képes leszel egy játékot fejleszteni”, vagy „jobban megismered a számítógépek működését” [14].

A tábor központi elemének a játékot választottuk. Munkánk során tapasztaljuk, hogy a tanulók rendszerint kapcsolatot vélnek felfedezni a videójátékok és az informatika tudásuk között. Logikájuk szerint „én sokat játszom videójátékokkal, ezért jó vagyok informatikából”. Természetesen a kettő között az esetek többségében semmilyen összefüggés nincsen. Ettől függetlenül tény, hogy az érdeklődő tanulók jellemzően szeretnek játszani. A diákok játékok iránti szeretetét alátámasztja a tábor első napján kitöltött kérdőív is (1. ábra). Az elképzelésünk, hogy ebből a lelkesedésből motivációt kovácsolunk, aztán informatikát tanítunk és számítógépes gondolkodást fejlesztünk a videójátékok fejlesztésén (game development), illetve a játék tervezés (game design) témakörökön keresztül.



1. ábra: a tanulók 65%-a jelentős időt fordít játékokra (1=egyáltalán nem jellemző, 6=teljesen jellemző).

2. Tábor feladatai, felépítése

A tábor 5 nap alatt került megrendezésre. Minden nap 9:00-tól 16:00-ig voltak 60 perces foglalkozások, egy-egy ebédszünettel. A beosztás tekintetében két lehetőségünk volt: vagy egy területre koncentrálnunk egy nap és csak abban a témában tartunk órákat, vagy minden nap több témát érintünk, ami miatt jóval kevesebb idő jut naponta egy témára. Mivel az utóbbit választottuk, abban bízva, hogy a diákok figyelme, érdeklődése jobban kitart, ha változatosabbak a programok. Ezzel a megoldással az egyes informatikai témakörök közötti tudástranszfer-elemek aktiválására is lehetőségünk nyílt.

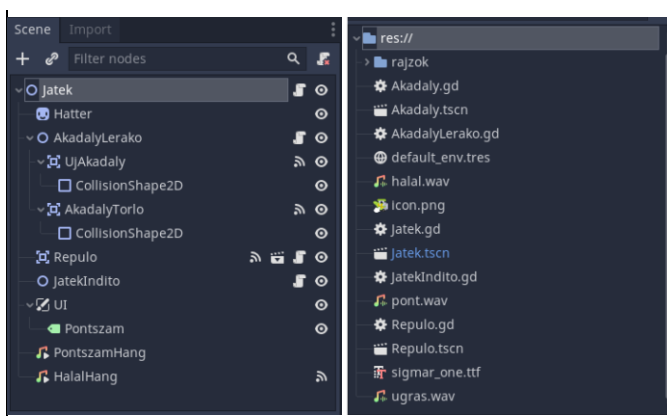
2.1. Játékfejlesztés

A tábor egyik központi témaköre a játékfejlesztés volt. Sajnálatos módon a középiskolákban ez nem egy kifejezetten népszerű módszer az informatika oktatására. Ezt alátámasztja a tény, hogy kevés olyan irodalom, kutatási anyag elérhető, amely a játékfejlesztés középiskolai alkalmazásainak lehetőségét taglalja [11]. Bizakodásra ad okot azonban, hogy az utóbbi néhány évben egyre több ilyen kutatás jelenik

meg. Viszont ezeknek jelentős része nem a szöveg alapú programozással, hanem a vizuális, blokk-alapú nyelvekkel foglalkozik [5][18].

Ettől függetlenül számos előnye van a játékfejlesztés iskolai alkalmazásának. Az egyik az interaktivitás. A tábor ideje alatt két kisebb projektet készítettünk el. Mindkét feladat elején megmutattuk a végleges verzió demóját, mielőtt elkezdtük a munkát. A tanulók rendkívül motiváltak lettek a végeredmények láttán. A hagyományos szövegalapú programozáshoz tartozó inerszegényebb integrált fejlesztői környezetekkel szemben az audiovizuális megjelenés és az interaktivitás jóval motiválóbb hatást gyakorolt a diákokra. Egy másik nagy előnye, hogy könnyű hosszabb távú célokat kitűzni a diákoknak, hiszen egy egyszerűbb játék elkészítése is komplex feladat. Ez is pozitív hatást gyakorol a tanulók motivációjára [14].

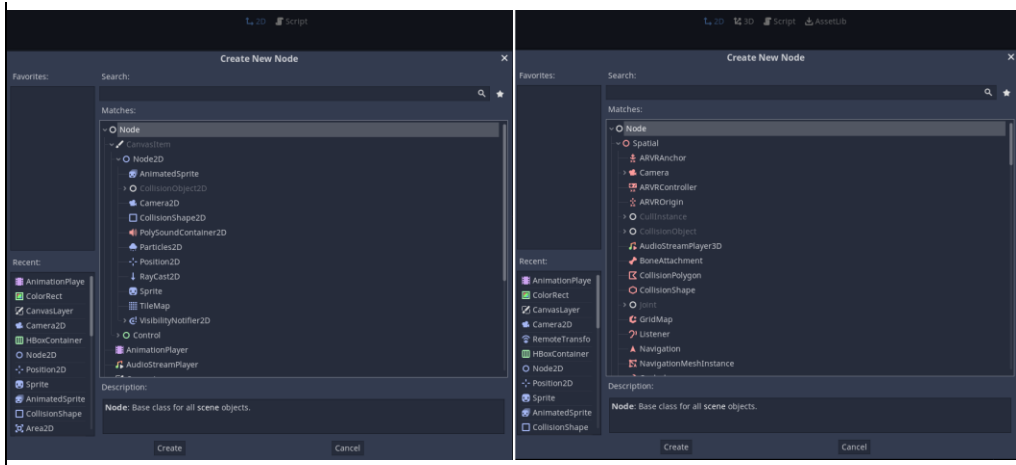
Egy kis kitérével érdemes megemlíteni, hogy munkánk során rendszeresen tapasztalunk egy érdekes jelenséget. A diákoknak egyre nagyobb nehézséget jelent megérteni, hogy miért van szükség a számítógépes munka során a helyes fájlnemek megválasztására és a fájlok könyvtárakba történő rendszerezésére [15]. Ennek oka véleményünk szerint az egyre felhasználóbarátabb telefonos operációs rendszerekben keresendő. Vegyük például a fényképek elkészítését és megtekintését. A legnépszerűbb rendszerek Galéria alkalmazásai képesek dátum, helyszínek vagy akár emberek alapján is sorba rendezni, rendszerezni a képeket anélkül, hogy a felhasználónak bármit kellene csinálnia. Ez a mindennapi életben hasznos és kényelmes, de sajnos a számítógépes munka során ez sokszor komoly gondokat okoz. A játékfejlesztés két okból is alkalmas ennek a tanítására. Egyrészt a játékban létrehozott objektumokat is muszáj beszédesen elnevezni és rendszerezni, ha nem akarunk elveszni a játék fejlesztése során. Másrészt ez a játék objektumain kívül ugyanúgy igaz a projekthez tartozó állományokra is. Így két helyen is gyakoroltathatjuk a diákokkal a helyes fájlnelvezést és az állományok rendszerezését (2. ábra).



2. ábra: beszédes objektumnevek (bal),
és helyesen elnevezett fájlok (jobb).

A játékfejlesztéshez az ingyenes és nyílt forráskódú Godot Engine játékmotort használtuk [6]. Ez a program minden tekintetben tökéletes iskolai használatra: a legutóbbi stabil verzió fájlmérete 72 MB, ezért könnyen és gyorsan beszerezhető. Telepíteni sem szükséges, így nincs probléma a rendszergazdai jogosultságokkal. A kis fájlméret mellett nagyon alacsony hardverigényekkel rendelkezik, így a korszerűtlenebb, régi gépeken is gond nélkül fut. Ezen kívül a játékok programozhatók több nyelven is. Az alapértelmezett választás a GDScript, ami a saját interpreteres nyelve a motornak. Ez a nyelv a leginkább kezdőbarát választás, hiszen nagyon hasonlít a Python nyelvre. Ezen kívül lehetőség van a kódolásra C, C++ és C# nyelveken is [6]. Végül egy kifejezetten oktatáshoz hasznos beállítás, hogy

elrejtjük a szerkesztő bizonyos funkcióit. A táborban például a teljes 3D részt kikapcsoltuk. Ez rengeteget segít a diákoknak, hogy a felhasználói felület ne tűnjön elsősorban túl ijesztőnek (3. ábra).



3. ábra: kezdőknek szánt profil, csak a 2D elemekkel (bal), valamint az alapértelmezett profil, benne minden 3D, VR eszközzel is (jobb).

2.1.1. Ismerkedés az alapokkal

Az első elkészített „játék” egy bevezető feladat volt, aminek a célja, hogy a diákok megismerkedjenek a játékmotor sajátosságaival. Erre azért volt szükség, mert minden felsorolt előnye ellenére a játékfejlesztés egy komplex feladat. A ma rendelkezésre álló játékmotorok nagyban megkönnyítik a helyzetet, de ez természetesen többlet feladattal is jár. A tanulóknak ugyanis meg kell ismerkedniük a játékmotor felépítésével, működési elvével is.

A példa működése nagyon egyszerű: a diákok előre leraknak szilárd falakat a világba, bal kattintásra pedig „bedobnak” egy labdát. A labdákra hat a gravitáció, a falak pedig megállítják a labdákat. A projekt tökéletes volt arra, hogy a tanulók megismerkedjenek a Godot Engine működésével. Ezt a feladatot a játékmotor dokumentációja ihlette, ahol található egy nagyon hasonló program, a példányosítás (instancing) fejezetben [6]. Természetesen a legtöbb diák kihasználta a lehetőséget, és tesztelte, hogy a rendszer hány labdát bír megjeleníteni mielőtt megadja magát (4. ábra). A program forráskódja elérhető, szabadon felhasználható bármilyen célra [8].

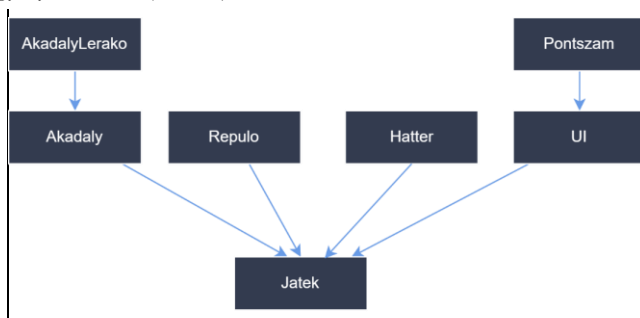


4. ábra: a diákok teletöltik labdákkal a „medencéjüket”.

2.1.2. Zuhanó repülés játék

A játékfejlesztés modul fő feladata egy Flappy Bird játék elkészítése volt. A játékot alkalmasnak találtuk kezdőprojektnek, hiszen a szabályrendszere elég egyszerű. Ennek ellenére viszont egy szórakoztató, akár mobiltelefonnal is kompatibilis, annak idején kifejezetten népszerű játékról van szó [21]. A választás utólag jó döntésnek bizonyult, a gyerekek élvezték a feladatot, elmondásaik alapján ez volt az egyik kedvenc részük a táborban. Egyébiránt a Harvard egyetem szabadon elérhető *CS50: Introduction to Game Development* nevű játékfejlesztés kurzusában is az első játékok között szerepel [2].

A Godot Engine előnyei közé tartozik még, hogy a működési filozófiájának köszönhetően könnyedén meg lehet tervezni a játék felépítését. Ha egyszerű ábrákkal leírjuk a legfontosabb objektumokat és azok kapcsolatát, gyakorlatilag meg is terveztük a játék felépítését. A diákok percek alatt össze tudtak rakni egy-egy ilyen tervet (5. ábra).



5. ábra: fontosabb objektumok a Zuhanó repülés játékban.

A játék fejlesztése során a tanulók megismerkedtek a változókkal, elágazásokkal és a ciklusokkal is. A részfeladatokat, fejlesztési fázisokat úgy választottuk meg, hogy a kognitív túlterhelés elkerülése érdekében egy foglalkozás során mindig csak egy új programozási eszköz kerüljön bevezetésre. Fontos volt ezen kívül, hogy a megtanult eszközök használatát gyakoroltassuk is velük a fejlesztés során. Ha délután már fáradtabbak voltak a tanulók, akkor a rajzok, képek beállításával és játékteszteléssel dolgoztak (6. ábra).



6. ábra: a repülő objektum beállítása Godot Engine-ben.

A diákok által elkészített játékok elérhetők, böngészőprogramból is kipróbálhatók az iskola itch.io oldalán keresztül [3]. A játék forráskódja elérhető, szabadon felhasználható bármilyen célra [9].

2.1.3. Hangrögzítés, hangszerkesztés

A tábor során terveztünk egy multimédiás foglalkozást is. Biztosítottunk a tanulóknak stúdiómikrofont és keverőpultot, így kipróbálhatták a hangrögzítést, hangszerkesztést és a hangkeverést is. Természetesen itt is konkrét feladat elé állítottuk őket. A Zuhanó repülés játékhoz kellett hangeffekteket készíteniük.

Tanulónként legfeljebb 3 hangeffektet kellett elkészíteni: az ugrás hangját, a pontszerzés hangját, valamint a repülő „felrobbanásának” a hangját. Bízattuk a kreativitásra a diákokat, és nem okoztak csalódást: volt, aki kulcsot, kulacsot vagy épp a cipőjét használta a hangok rögzítéséhez, de legtöbben persze a saját hangszállaikra hagytak. A rögzített hangokat az ingyenes és nyílt forráskódú Audacity nevű szoftverben rögzítettük, szerkesztettük [1].

Érdekes tapasztalat, hogy a diákok maguktól nem igazán gondolnak rá, hogy az előzetesen megszerzett számítógépes ismereteiket számukra eddig ismeretlen szoftverekben is kipróbálják, kamatoztassák. Ez abban nyilvánult meg, hogy kezdetben nehezen boldogultak az Audacity felületén. Viszont meglepő módon semmi problémájuk nem volt a szoftver kezelésével miután megjegyeztük, hogy az általuk ismert másolás, kivágás, beillesztés, törlés, kijelölés műveletek ebben a programban is ugyanúgy működnek.

Voltak olyan félénk diákok is, akik ódzkodtak a mikrofon használatától vagy éppen nem szerették volna visszahallani a saját hangjukat. Nekik biztosítottunk arra lehetőséget, hogy a Labchirp nevű, szintén ingyenes és nyílt forráskódú programmal generálhassanak nekik tetsző hangeffekteket [12].

2.2. Játéktervezés

Egy másik oszlopos témaköre a tábornak a játék tervezés (game design). Ez egy nehezen körbehatárolható terület, hiszen nemcsak játékok tervezésével foglalkozik, hanem „játékosok tapasztalatainak megalkotásával” is [10].

A foglalkozások során három témakört jártunk körbe a diákokkal:

- az érzelmek és a játékok kapcsolata,
- a játékelmény,
- a játékok célrendszere.

Ezekben nagy segítségünkre volt az oktatóknak ingyenesen felhasználható, Riot Games által kiadott játéktervezői kurzus, az *Urf Academy* [19]. Ennek a tananyagának a moduljaiból indultunk ki, végül kisebb-nagyobb módosításokkal használtuk.

A játéktervezéses foglalkozások során kizárólag csoportmunkában dolgoztak a tanulók és szinte kivétel nélkül a felfedezéses tanulás módszerére hagytak. A tanulói csoportokat tudatosan alakítottuk ki, minden csapatot egy-egy szint képviselt. Az első feladatuk az volt, hogy gyűjtsék össze, majd csoportosítsák, hogy a játék (bármilyen játék) milyen érzelmeket válthat ki a játékosokból (7. ábra).



7. ábra: érzelmek és játék kapcsolata.

A következőkben arról beszélgettünk, hogy mit jelent az, hogy egy játék szórakoztató (fun). A *szórakoztató* fogalom tervezői szemmel nézve haszontalan kifejezés, hiszen nem elég konkrét. Emiatt érdemes megismerkedni a Hunicke, LeBlanc & Zubek féle csoportosítással, mely szerint egy játék 8 féle módon lehet szórakoztató [20]. Miután ezekre mindenki tudott saját példákat mondani, áttértünk a játékok céljaira.

A diákok 4 fős csoportokban dolgoztak. A két összetolt asztal 4 oldalának a közepére mindenki felragasztott egy kaput. Ezután papírfocilabdát kellett hajtogatniuk, majd kezdődhetett a játék. A játékszabályok egyszerűek:

- A játék úgy kezdődik, hogy minden labda az asztal közepén van.
- Ha bármelyik labda leesik, utána az asztal közepére kerül vissza.
- A játék CÉLJA, hogy a labdával a másik játékos labdáját a saját kapujukba juttasd (meg is kell állnia ott).
- Minden játékosnak 3 lépése van egy körben.
- A labdát pöckölni, vagy suhintani lehet.
- A játékos köre véget ér, ha bármelyik labda leesik az asztalról.

Az első játéktesztelés célja, hogy a tanulók megismerjék a kiindulási állapotot, teljesen megértsék a játékszabályokat (8. ábra).



8. ábra: papírfoci játék.

Ezután következett az igazán érdekes része a feladatnak. A tanulók kitaláltak különböző célokat a játékhoz, a szabályok változtatása nélkül. Ezekkel kísérleteztek, tesztelték, hogy a cél megváltoztatásával hogyan változik a játékelmény. Kerestük a legszórakoztatóbb célokat, megbeszéltük, hogy mitől jó vagy rossz egy cél a papírfoci játék esetében.

Érdekességként érdemes megemlíteni, hogy az *Urf Academy* oktatóknak szóló leírásában részletezi, hogy az Egyesült Államok oktatási rendszerében megnevezett fejlesztendő attitűdök, alapkészségek közül melyikeket fedi le a képzés. Ilyen például a kooperációs képesség, íráskészség, beszédképesség, önkifejezés fejlesztése, egy szakterület területspecifikus szókincsének magabiztos használata vagy akár komplex valós problémák megoldásának megtervezése, a feladat több, kisebb problémára bontásával [19]. Ezek nagyon hasznos 21. századi képességek bármelyik diáknak, függetlenül attól is, hogy milyen munkakörben fognak a későbbiekben elhelyezkedni.

2.2.1. Társasjátékozás

A játéktervezést összekötöttük társasjátékozással is. Ennek a szórakozáson és a kikapcsolódáson felül az volt a célja, hogy tervezői szemmel is vizsgálják meg az egyes játékokat. Minden csapat rotációban próbálta ki a stílusukban nagyon különböző játékokat, majd ki kellett tölteniük egy elemző lapot a társasjátékokra vonatkozóan.

Ezzel lehetőséget biztosítottunk a diákoknak arra, hogy kamatoztassák a játéktervezés foglalkozásokon megszerzett ismereteiket. Képesek voltak elemezni a játékokat, eldöntötték közösen, hogy mi teszi szórakoztatóvá vagy épp frusztrálóvá. Ezen felül írhattak arra vonatkozóan ötleteket, hogy hogyan lehetne változtatni, esetleg javítani a játékon. Itt főleg a célok megváltoztatására helyeztük a hangsúlyt, hiszen arról tanultak a legtöbbet.

2.3. Unplugged informatika

A táborban minden nap 6 órányi foglalkozáson vettek részt a tanulók. Az egészségükre is gondolnunk kellett: nem ülhetek minden nap hat órán keresztül egy kényelmetlen székben és nem is tölthettek ebből az időből négy órát a képernyő előtt.

Az unplugged (számítógép használata nélküli) informatikai foglalkozások bevezetésével erre nem is volt szükség. A *CS Unplugged* programjai közül válogattunk témákat. Ezek olyan feladatok, tananyagok, amelyek kézzel fogható játékokon keresztül mutatnak be különböző számítógéptudományi témákat [22].

A tanulók változó informatikai háttérrel rendelkeztek, ezért olyan feladatokat választottunk, amelyek bevezető témának alkalmasak, de mindenki számára biztosítanak új ismereteket:

- bináris számrendszer, előjel nélküli egész számok ábrázolása,
- bevezetés a digitális képábrázolásba, faxgép működése,
- tömörítési eljárások, veszteséges és veszteségmentes tömörítés.

A feladatokat „magyarosítottuk”, illetve kisebb-nagyobb módosításokkal, extra gyakorló példákkal kiegészítve használtuk őket [22]. Mindegyik témát diskusszióval zártuk, megbeszéltük, hogy a mindennapi életünkben hol és hogyan fedezhetők fel ezek az informatikai jelenségek.

A diákok nagyon élvezték, hogy egy játékfejlesztés foglalkozás után kimehettek az udvarra és a nyári napsütésben „piknikeztek” a feladatok megoldása közben (9. ábra).



9. ábra: ismerkedés a tömörítési algoritmusokkal a szabadban.

2.3.1. Drón programozás

Az unplugged foglalkozásokat egy meglepetés feladattal zártuk. A diákoknak csak annyit árultunk el a feladattal kapcsolatban, hogy pármunkáról lesz szó. Az iskolánkban rendelkezésre állt egy Python nyelven programozható drón.

A feladat két részből állt. Először a tanulók kiválasztottak egy pályakártyát a pakliból. A pálya tanulmányozása után a kezükbe vettek egy mérőszalagot és megépítették az akadálypályát úgy, hogy az alakzat minden sarkára elhelyeztek egy lesúlyozott héliumos lufit (10. ábra).



10. ábra: megépített akadálypálya lufikkal (bal), laptop, drón és mérőszalag (jobb).

A pálya megépítése után tértek át a programozásra. Az egyikük megfordította a pályakártyát, aminek a hátulján megtalálható a drón utasításkészlete. A másik diák gépelte a forráskódot. A párosnak együtt kellett megterveznie a program algoritmusát, a „kártyás” ember pedig segítette a gépelő társát a szintaxissal. Érdeemes megjegyezni, hogy a drónhoz alapértelmezetten tartozó SDK parancsértelmezőként működik, azaz egyesével lehet a drónnak utasításokat küldeni hálózaton keresztül.

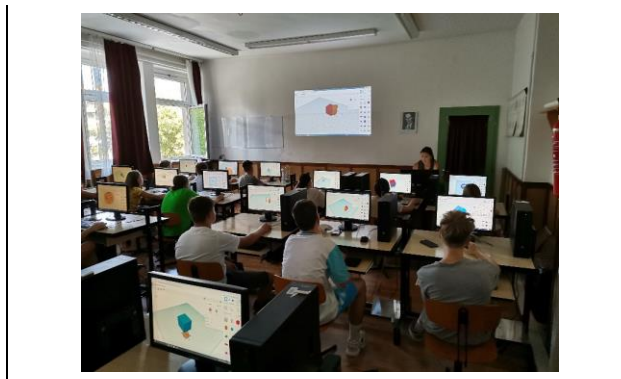
Ezért írtunk egy csomagoló API-t, amely segítségével előre megírható a drón teljes útvonalának programkódja. A program szekvenciálisan, másodpercenként küldi át az utasításokat. Így egy LOGO-szerű program írásával lehetett programozni a drón mozgását. A kék lufik közé becsempésztünk egy rózsaszínt is. A diákoknak kiadtuk az utasítást, hogy amikor a jármű a rózsaszín lufi felé ér, akkor szaltóznia kell egyet.

A tanulók személyes beszámolója alapján ez a meglepetés feladat volt az egyik kedvencük a táborban. A pályák és a csomagoló API szabadon felhasználható és ingyenesen elérhető a feltüntetett hivatkozáson keresztül [7].

2.4. 3D modellezés és nyomtatás

Végül a tábor utolsó témaköre következett, a 3D modellezés és a 3D nyomtatás. A tervezéshez a böngészőből elérhető, kezdőbarát és teljesen ingyenes Tinkercad programot használtuk [24]. A tanulóknak lehetősége volt a matematika órákon halmazok és geometria témakörökben megszerzett tudásukat kamatoztatni.

Az első feladat egy olyan kulcstartó tervezése volt, amely formája megegyezik az iskola címerével. A diákok nagy lelkesedéssel, motivációval vetették bele magukat ebbe a témakörbe is (11. ábra).



11. ábra: 3D modellezés szárnypróbálgatások.

Az első modell elkészítése jóval gyorsabban ment a tervezetthez képest. Így maradt idő a foglalkozáson arra, hogy mindenki megtervezze a saját kulcstartóját, amit ki is nyomtathat a tábor ideje alatt. A tanulók kreatívan oldották meg a feladatot, nagyon szép modellek, kulcstartók születtek (12 ábra).

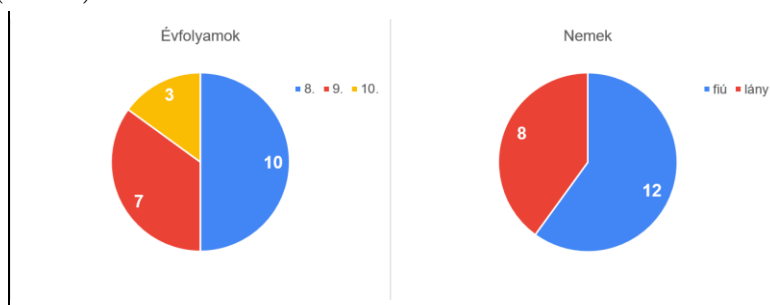


12. ábra: az egyik diák által készített baglyos kulcstartó.

3. Tapasztalatok, következtetések

Összességében elmondható, hogy a tábor elérte a célját. A diákok személyes beszámolója alapján jól érezték magukat, sok újat tanultak és gyorsan eltelt nekik a hét. Tanári oldalról azért érezhető volt, hogy sok foglalkozást szerveztünk nekik, az utolsó napokban már fáradtak voltak, valamint egyre nehezebben tudtak koncentrálni a délutáni órákban. Fontos megjegyezni azt is, hogy pályafutásunk során először szerveztünk ilyen tábort, így a végeredményhez hozzátartozik, hogy nekünk is bőven van miben fejlődni, tapasztalatot gyűjteni.

A táborban összesen 20 tanuló vett részt, akiknek évfolyam- és nem szerinti eloszlását mutatja az alábbi ábra (13. ábra).



13. ábra: a tanulók aránya évfolyam (bal) és nem szerint (jobb).

A résztvevőkkel kitöltöttünk egy névtelen kérdőívet a tábor első napjának első foglalkozásán, illetve lezárásként a foglalkozások befejeztével a hét legvégén is.

Mindenképpen érdemes összehasonlítani két kérdést, ahol a diákok saját szavaikkal rövid szöveges választ adtak:

- Azért jelentkeztem a táborba, mert... (tábor elején)
- Leginkább az tetszett a táborban, hogy... (tábor végén)

Az első kérdésre 10 válaszadó indokolt azzal, hogy „szeretem / érdekel az informatika”, 3 konkrétan a programozást emelte ki, 3 a 3D nyomtatást nevezte meg, 3 másik pedig egyéb indokokat sorolt fel.

Ezzel szemben a tábor utáni kérdésben 10 válaszadónak a játékfejlesztés volt a kedvence, 8-an pedig a jó hangulatot, a társaságot és a kreatív, változatos programokat emelték ki.

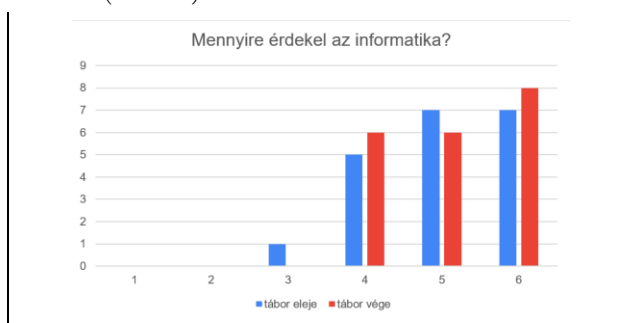
Ebből azt a következtetést vonhatjuk le, hogy a programterv előzetes ismertetése ellenére a diákoknak nem volt elképzelésük arról, hogy milyen programokra, konkrét feladatokra számíthatnak a foglalkozások nevei alapján. Örömmel vettük, hogy a tábor végére képesek voltak egyértelműen kiemelni a számukra legfontosabb előnyöket. A tény, hogy a válaszok fele a játékfejlesztést emelte ki, mindenképpen bizakodásra ad okot a játékfejlesztés témakör oktatási felhasználásra.

A tábor elsődleges célja a diákok motiválása volt az informatika tanulásához. Tekintve, hogy pályakezdő pedagógusként először, előzetes tapasztalatok nélkül szerveztünk ilyen programot, nem célunk bizonyítani, hogy ez a módszer hatékony a motiváció növelésére. A kérdőíveket a saját tapasztalataink rendszerezésére, az eredmények könnyebb értelmezése érdekében készítettük. Ez nagyban elősegíteni a jövőbeli munkánkat.

A tábor elején és végén is megkérdeztük tőlük a következőket:

- Mennyire érdekel az informatika? (1–6 Likert-skála)
- Mennyi motivációd van az informatika tanulásához? (1–6 Likert-skála)

Az első kérdésben minimális eltérés mutatkozott a tábor előtt és után, a diákok többségét kifejezetten érdekli az informatika (14. ábra).



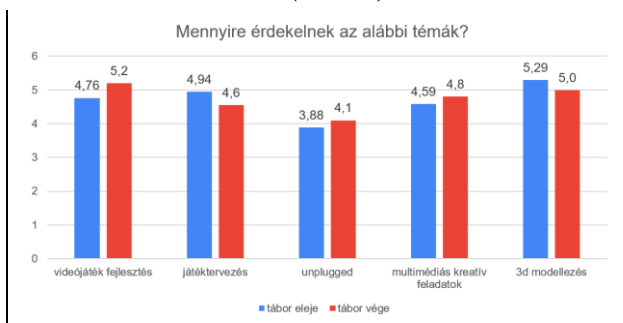
14. ábra: első kérdésre adott válaszok a tábor elején (kék) és végén (piros).

A motivációra vonatkozó kérdésben viszont már jelentősebb eltérés mutatkozik a két kitöltés között (15. ábra). Pozitív irányú elmozdulások jelennek meg, a korábban ötös értékelést adó diákok közül a tábor befejeztével sokan a hatos választ („rendkívül motivált vagyok”) jelölték meg. A másik végletet nézve pedig eltűntek a kettes értékelések, amelyek már erősen a másik véglet felé húztak („szinte nincs motivációm, nagyon unalmasak az órák”).



15. ábra: második kérdésre adott válaszok a tábor elején (kék) és végén (piros).

Érdekes összehasonlítási alap, hogy a diákok saját bevallásuk szerint mennyire érdeklődnek egy-egy témakör iránt. A tábor elején és végén is megkértük őket, hogy értékeljék 1–6-ig a témákat. Ezeknek az átlagos pontszámait hasonlítottuk össze (16. ábra).



16. ábra: átlagos pontszámai az egyes témaköröknek a tábor elején (kék) és végén (piros).

Ez a diagram is alátámasztja a tanulók válaszait a tábor végén, miszerint a játékfejlesztés volt a kedvenc témakörük. Az átlagpontszámokban a legnagyobb pozitív növekedés ennél a témánál figyelhető meg és a tábor végén összességében is a játékfejlesztésé a legmagasabb értékelés. Ebből arra következtettünk, hogy az előzetes elvárásaikat is felülmúlta ez a témakör. Érdekes módon a 3D modellezés iránti érdeklődésük csökkent, pedig a tábor elején a pontszám is rendkívül magas volt, illetve a szöveges kérdésben is többen megemlítették ezt a témát.

A legnagyobb csökkenés a játéktervezés témakörben figyelhető meg, 3 tanuló is megemlítette szöveges válaszaiban, hogy unalmasnak találták. A saját észrevételünk alapján a papírfocira vonatkozó feladatot már élvezték a csoportok, így valószínűleg a bevezető foglalkozásokat lenne érdemes átdolgozni, élvezetesebbé tenni számukra. Ha több idő jutott volna játéktervezésre, valószínűleg gyökeresen más véleményük lenne róla, hiszen a kurzus végén minden csoportnak egy saját társasjátékot kell terveznie és tesztelnie. Összességében ez a visszajelzés nagyon hasznos volt, a jövőben mindenképpen újra fogjuk gondolni ezt a témakört.

A legalacsonyabb értékelést mind a két mérés során az unplugged foglalkozások kapták. Ez érdekes módon nem tükrözi a valóságot. 1 válaszadó ugyan kiemelte, hogy szerinte az unplugged feladatok unalmasak voltak, de a tábor ideje alatt a tanulók nagyon élvezték, hogy tantermen kívüli programokon is részt vehettek. Két különböző kérdésre összesen 7 olyan válasz is érkezett, amely kevesellte kinti programokat. Ezek alapján elmondhatjuk, hogy igénylik a diákok azt, hogy egy informatika táborban

sem egész nap a számítógép előtt üljenek. A személyes meglátásunk is az, hogy élvezték a kinti unplugged foglalkozásokat és a kötetlenebb beszélgetéseket.

Mindent egybevetve elmondhatjuk, hogy a diákok jó élményekkel, új tudással gazdagodva zárták az egyhetes élményinformatikai táborunkat. A szervezéssel és tervezéssel járó rengeteg kihívás ellenére pedagógusként is értékes tapasztalatokra, élményekre tehetünk szert. Bízunk benne, hogy a résztvevő diákok informatika tantárgy iránti motivációjára valóban pozitív hatást gyakorolt a tábori részvétel.

Irodalom

1. Audacity: *Free, open source, cross-platform audio software for multi-track recording and editing*. <https://www.audacityteam.org/>. (utoljára megtekintve: 2022. 10.)
2. C. Ogden, D. J. Malan: *CS50's Introduction to Game Development, Flappy Bird*. (2018). <https://cs50.harvard.edu/games/2018/weeks/1/>. (utoljára megtekintve: 2022. 10.)
3. Diósgyőri Gimnázium: *DIN 2022 játékok*. (2022). <https://diosgyori.itch.io/>. (utoljára megtekintve: 2022. 10.)
4. D. Kahneman: *Thinking, Fast and Slow*. New York: Farrar, Straus; Giroux. (2011).
5. G. Csapó: *Placing Event-Action-based Visual Programming in the Process of Computer Science Education*. APH. 16 (2), 35-57. (2019).
6. Godot Engine: *Documentation*. (2022). <https://docs.godotengine.org/en/stable/>. (utoljára megtekintve: 2022. 10.)
7. Gulácsi Ádám: *Github – drón csomagoló API forráskódja*. https://github.com/guladam/dji_tello_edu_py. (utoljára megtekintve: 2022. 11.)
8. Gulácsi Ádám: *Github – játékefejlesztés bevezető projekt forráskódja*. (2022). https://github.com/guladam/din_2022_labda_godot. (utoljára megtekintve: 2022. 10.)
9. Gulácsi Ádám: *Github – játékefejlesztés zuhanó repülés projekt forráskódja*. (2022). https://github.com/guladam/din_2022_zuhano_repules_godot. (utoljára megtekintve: 2022. 10.)
10. J. Schnell: *The Art of Game Design: A Book of Lenses*. Taylor & Francis Group. (2019).
11. K. M. L. Cooper, W. Scacchi: *Computer Games and Software Engineering*. Taylor & Francis Group. (2020).
12. Labchirp: *A Free and versatile sound effects generator*. <http://labbed.net/software/labchirp/>. (utoljára megtekintve: 2022. 10.)
13. Magyar Közlöny: *2020 évi 17. szám*. (2020). <https://magyarkozlony.hu/dokumentumok/3288b6548a740b9c8daf918a399a0bed1985db0f/letoltes>. (utoljára megtekintve: 2022. 10.)
14. M. K. Alderman. *Motivation for achievement: Possibilities for teaching and learning*, Routledge. (2013).
15. M. Csernoch: *The Stepchild of Informatics Education: File Management*. Academia Letters 2021 1-4 (2021).
16. M. Csernoch: *Thinking Fast and Slow in Computer Problem Solving*. Journal of Software Engineering and Applications, (2017) 10(1). http://file.scirp.org/pdf/JSEA_2017012315324696.pdf. (utoljára megtekintve: 2022. 10.)
17. M. Csernoch, P. Biró, J. Máth, K. Abari: *Testing algorithmic skills in traditional and non-traditional programming environments*. Informatics in Education. 14 (2), 175-197, 2015.
18. P. Domokos, M. Széll, V. Takács: *Blocklino: a graphical language for Arduino*. In: 8th IEEE International Conference on Cognitive Infocommunications: CogInfoCom 2017 : Proceedings : September 11-14, 2017 Debrecen, Hungary, IEEE Computer Society, Piscataway, 45-50 (2017).
19. Riot Games: *Urf Academy*. (2022). <https://www.riotgames.com/en/urf-academy/curriculum-guide>. (utoljára megtekintve: 2022. 10.)
20. R. Hunicke, M. LeBlanc & R. Zubek. *MIDA: A formal approach to game design and game research*. Proceedings of the AAAI Workshop on Challenges in Game AI (Vol. 4, No. 1, p. 1722). (2004).

21. R. Syngel: *The rise and fall of Flappy Bird*. (2014).
<https://www.wired.co.uk/article/flappy-bird>. (utoljára megtekintve: 2022. 10.)
22. T. Bell, I. Witten: *CS Unplugged: An enrichment and extension programme for primary-aged students*. (2015).
<https://www.csunplugged.org/en/>. (utoljára megtekintve: 2022. 10.)
23. T. Nagy, M. Csernoch, P. Bíró: *The Comparison of Students' Self-Assessment, Gender, and Programming-Oriented Spreadsheet Skills*. *Education Sciences*. 11 (10), 1-29, 2021.
24. Tinkercad: *3D digitális tervek létrehozása online CAD-del*.
<https://www.tinkercad.com/>. (utoljára megtekintve: 2022. 10.)