

Algoritmikus és alkalmazói készségek tesztelése

Csernoch Mária, Biró Piroska

{csernoch.maria, biro.piroska}@inf.unideb.hu

Debreceni Egyetem Informatikai Kar

Absztrakt. A 2011/2012-es tanév szeptemberében a Debreceni Egyetem Informatikai Karán útjára indítottuk az Algoritmikus és Alkalmazói Készségek Tesztelése című projektet. Átfogó képet szerettünk volna kapni arról, hogy az Informatikai Karon tanulóit milyen ismeretekkel rendelkeznek, és hogyan tudják ezt a szakma nyelvén megfogalmazni. Azt vizsgáltuk továbbá, hogy milyen szintű algoritmikus készségekkel rendelkeznek a hallgatók, az alkalmazói szoftverek használata során mennyire fejlődött/fejlődik az algoritmikus készségük. A kutatásaink további célja, hogy eredményeinket összevessük más korábbi mérések eredményeivel és magyarázatot adjunk a különböző mérésekből származó esetleges eltérésekre. Az összehasonlító elemzéseink további forrása az érettségi eredmények, valamint a projekt keretein belül végzett tanári- és önértékelések.

Mottó: Miért leszek informatikushallgató?

Mit tudok informatikából?

Mit hiszek, hogy tudok informatikából?

Mit hisznek a tanárok, hogy mit tudok informatikából?

1. Bevezetés

A 2011/2012-es tanév szeptemberében indítottuk útjára az Algoritmikus és Alkalmazói Készségek Tesztelése (AÉAKT) című projektet. A projekt céljai között szerepel annak tesztelése, hogy az Informatikai Karon tanulóit milyen szakmai ismeretekkel rendelkeznek, hogyan tudják ezt a szakma nyelvén megfogalmazni, hogyan értékelik saját tudásukat, valamint azt, hogy informatikatanárok hogyan értékelik a hallgatók tudását. Mindezekon túl, átfogó képet szerettünk volna arról kapni, hogy mit jelent a hallgatók számára az informatika, miért választották ezt a tudományterületet.

A felmérést megelőzően már jelentek meg rémisztő hírek az informatikushallgatók alulteljesítéséről. Az egyik típusú, ilyen adatokat tartalmazó jelentések arról számolnak be, hogy az informatikushallgatók mindösszesen közel 50% halad és végez a mintatanterv szerint (1. táblázat). A Műszaki Kar hallgatóinak teljesítménye közelíti meg leginkább az informatikushallgatókét, minden más kar esetében sokkal kisebb mértékű a lemorzsolódás. Mi lehet a magyarázata az informatikusok és a műszaki hallgatók ilyen szintű alulteljesítésének? Jelen tanulmányban azt vizsgáljuk, hogy milyen az átmenet a középiskolai és a felsőoktatási intézmények között. Azt nézzük, hogy milyen ismeretekkel érkeznek a hallgatók, milyen elképzelésük van a tudományterületről, a szakmáról, valamint azt, hogy milyen készségek hiányoznak, amelyek feltétlenül szükségesek az informatikai tanulmányokhoz.

	2010		2011	
	Képzési idő alatt végzett	Képzési időn túl végzett	Képzési idő alatt végzett	Képzési időn túl végzett
ÁJK	79%	21%	76%	24%
ÁOK	75%	25%	98%	2%
BTK	78%	22%	83%	17%
IK	56%	44%	46%	54%
KTK	80%	20%	63%	37%
MK	57%	43%	46%	54%
TTK	71%	29%	61%	39%

1. táblázat: Képzési idő alatt és képzési időn túl végzett hallgatók aránya a Debreceni Egyetem kiválasztott karain 2010-ben és 2011-ben: ÁJK (Általános és Jogtudományi Kar), ÁOK (Általános Orvostudományi Kar), BTK (Bölcsészstudományi Kar), KTK (Közgazdaságtudományi Kar), IK (Informatikai Kar), MK (Műszaki Kar), TTK (Természettudományi Kar)

2. Minta

Az AéAKT projektet 2011 szeptemberében indítottuk, melynek keretein belül az első tesztet 2011 szeptemberében, a tanév megkezdésekor végeztük. A projekt három nagyobb szakaszra bontható, amelyeket megismételtünk a következő két tanév hasonló időszakaiban.

- szeptember
 - a középiskolából hozott ismeretek tesztelése,
- november
 - egy Computer-algorithmic and debugging-based (CAAD-alapú) módszer [10] táblázatkezelői környezetben történő tesztelése,
- következő év november
 - a CAAD-alapú módszer tesztelése egy évvel később.

A 2011/2012-es tanév szeptemberében a Debreceni Egyetem Informatikai Karának (DE IK) Programtervező informatikus (PTI), Mérnök-informatikus (MI), Gazdaságinformatikus (GI), Informatikus könyvtáros (IK) nappali, Informatikatanár (IT) és egyéb levelezős hallgatóit teszteltük. Következő év szeptemberében ugyanezeket a szakokon megismételtük a felmérést az újonnan érkezett elsőéves hallgatókkal. 2013 szeptemberében sikerült más egyetemeken és főiskolák informatikushallgatóival is elvégezni a felmérést. Csatlakozott a felméréshez az Eötvös Lóránd Tudományegyetem (ELTE), a Nyíregyházi Főiskola (NYF) és az egeri Eszterházy Károly Főiskola (EKF)¹ (2. táblázat). Megkerestük az ország több felsőfokú intézményét is, de a többiek nem válaszoltak vagy nem kívántak részt venni a felmérésben. Terveink között szerepel azon-

¹ Szervezési és adminisztrációs okok miatt az EKF hallgatói októberben írták meg a tesztet.

ban, hogy következő év szeptemberében az ország valamennyi informatikai képzést végző intézményében elvégezzük a felmérést.

	2011	2012	2013
DE_IK PTI	115	107	115
DE_IK MI	86	112	115
DE_IK GI	109	101	90
DE_IK IK	27	15	10
DE_IK IT	16	2	–
ELTE			325
EKF			72
NYF			66

2. táblázat: Az AéAKT projekt felmérésében részt vevő hallgatók száma szakonkénti és intézményenkénti megosztásban

3. Projekt tesztsjei

3.1. Szeptemberi felmérés feladatai

A szeptemberi felmérés feladatai hat kategóriába sorolhatók: (1) billentyűzet- és egérhasználat, (2) fájlkezelés és hálózati ismeretek, (3) pseudo kód megfejtése (1. ábra), (4) logikai műveletek (1. ábra), (5) táblázatkezelés (2. és 3. ábra), (6) szövegszerkesztés (4. és 5. ábra). Mivel mindhárom évben ugyanazokkal az elsődleges célokkal indítottuk a felmérést, ezért a feladatok jellege megmaradt, azonban a kitűzött feladatokon végeztünk változtatásokat. Jelen dolgozatban terjedelmi határok miatt azokat a feladatokat mutatjuk be, amelyek mindhárom évben szerepeltek. A logikai műveleteket, a pseudo kód megfejtését (1. ábra) és a táblázatkezelői elméleti ismereteket (3. ábra) tesztelő feladatok változatlanul szerepeltek valamennyi tesztben. A táblázatkezelői gyakorlati feladatokat 2013-ban kiegészítettük egy újabb kérdéssel (2. ábra), míg a szövegszerkesztő feladatok esetében ugyanazt a mintát mellékelve három különböző módszert és ezek kombinációit használtuk (4. és 5. ábra).

Egy játékban két kártyát (X, Y) húzunk egy-egy pakliból. Mindkét pakliban találhatók A-betűt, B-betűt és 0-s számjegyet tartalmazó kártyák. Add meg, hogy az alábbi algoritmus alapján milyen esetekben hány pontot kaphatunk!

	X	Y	Pont
V:="A" vagy Y="A"	A	A	
W:="B" vagy Y="B"	A	B	
Ha V és W akkor Pont:=3	A	0	
különben Ha V akkor Pont:=2	B	A	
különben Ha W akkor Pont:=1	B	B	
különben Pont:=0	B	0	
	0	A	
	0	B	
	0	0	

Mit csinálnak az alábbi programrészletek? Fogalmazza meg általánosan is, hogy a DB és az M változó mit tartalmaz!

N=50, a mért értékek:
500,500,500,500,500,600,600,650,700,750,820,880,930,1010,1050,980,930,830,780,720,720,710,700,
750,770,790,820,880,880,820,760,740,600,500,560,670,770,770,760,750,740,740,730,720.

DB:=0
a) Ciklus i=1-től N-ig
Ha X(i)>800 akkor DB:=DB+1
Ciklus vége

DB:=0
b) Ciklus i=2-től N-1-ig
Ha X(i)<X(i-1) és X(i)<X(i+1) akkor DB:=DB+1
Ciklus vége

M:=0
c) Ciklus i=2-től N-ig
Ha X(i)-X(i-1)>M akkor M:=X(i)-X(i-1)
Ciklus vége

1. ábra: A logikai műveletek (bal) és pseudo kód megfejtése feladatok (jobb). Mindkét feladatot a Nemes Tihmér Országos Informatikai Tanulmányi Verseny – Programozás kategória, 2008/2009-es tanév, 1. forduló, 1. korcsoport (5–8. osztály) feladatsorából választottuk [20], [21]

A táblázatkezelői ismeretek számonkérésére két különböző típusú feladatot adtunk. Az egyik feladat négy képlettel megoldandó kérdést (2. ábra a)–d) feladatok) és egy kódfejtést (2. ábra e) feladat) tartalmazott. 2013-ben a képlettel megoldandó kérdéseket kiegészítettük egy ötödikkel (2. ábra f) feladat), annak érdekében, hogy a novemberi felmérések során pontosabban tudjuk mérni, hogy a hallgatók a feladatok megoldása során milyen metakognitív döntéseket hoznak [1], [5], [8], [10], [12]. A másik táblázatkezelői feladatban négy függvény összehasonlítását kértük (3. ábra).

Az a)–d) és f) feladatokat Excel függvények felhasználásával oldja meg! A táblázat utolsó sora a 216. sor.

	A	B	C	D	E
1	ország	féldresz	fovaros	terület	fo_ezer_02
2	Afganisztan	Asia	Kabul	647500	27756
3	Albánia	Europe	Tirana	28748	3545
4	Algéria	Africa	Algiers	2381740	32278
5	American Samoa	Oceania	Pago Pago	199	69
6	Andorra	Europe	Andorra la Vella	468	68
7	Angola	Africa	Luanda	1246700	10593
8	Anguilla	Amerika	The Valley	102	12

- a) Írassa ki a legnagyobb területű ország fővárosát!
- b) Írassa ki az egyes országok népsűrűségét!
- c) Írassa ki az afrikai országok számát!
- d) Írassa ki azon országok átlagos területét, amelyek lakossága kisebb, mint egy, a H1 cellában tárolt érték!
- e) Mit csinál az alábbi összetett Excel képlet: $\{=SZUM(HA(B2:B216="Europe" * HA(BAL(A2:A216)="A",1))\}$?
- f) Írassa ki azon országok számát, amelyek területe nagyobb, mint egy, a G1 cellában tárolt érték!

2. ábra: A szeptemberi, a novemberi és a következő év novemberi táblázatkezelői kérdések (jobb) és a feladathoz tartozó minta táblázat (bal). Az f) kérdés nem szerepelt a 2011. és 2012. szeptemberi felmérésekben

A táblázatkezelői feladatok elméleti kérdésével tesztelni kívántuk, hogy a hallgatók ismerik-e a felsorolt függvények lehetőségeit, korlátait, mi alapján döntenek a használatukról. A gyakorlati feladatsor a) kérdése (2. ábra) ugyanezt az ismeretet kéri számon.

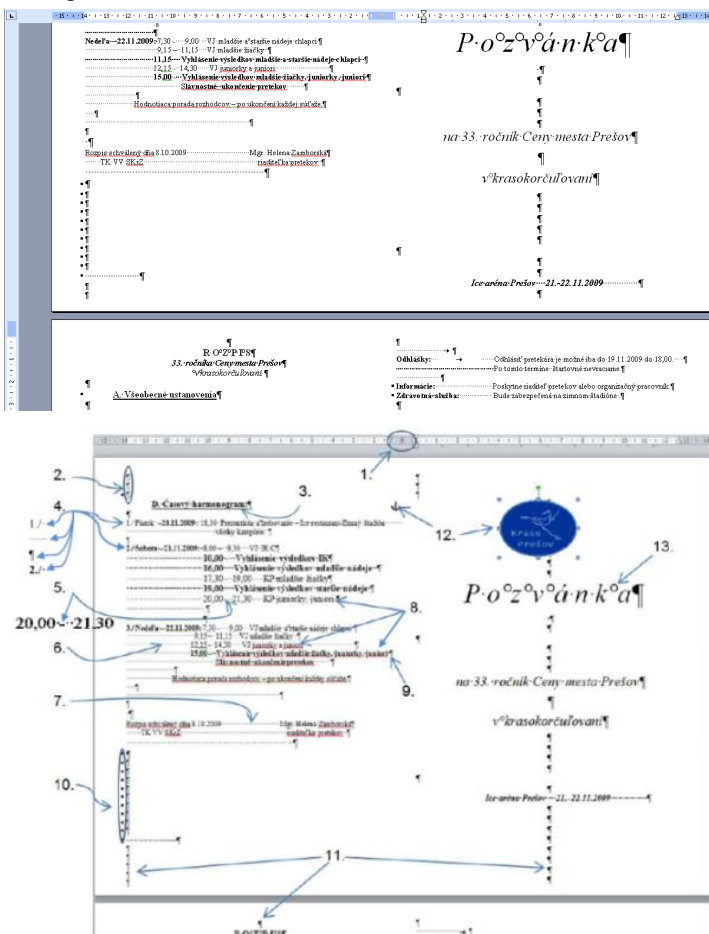
Hasonlítsa össze az INDEX és a HÖL.VAN függvényekből létrehozott összetett függvényt az FKERES és VKERES függvényekkel!

3. ábra: A táblázatkezelői feladatok elméleti kérdése [23]

Szövegszerkesztési ismeretek számonkérése csaknem kizárólagosan számítógépes környezetben történik, a papíralapú számonkérés módszerei jelenleg még nem kidolgozottak. Olyan feladatokat kellett tálalni, amelyek nem a menühasználatot kérik számon, hanem azt, hogy az adott szituáció milyen szövegszerkesztési problémát jelent [3], [5]. A szövegszerkesztési prob-

léma/problémák megjelenítéséhez képernyőmásolatot készítettünk a dokumentumokról, amelyek a szövegen túl a nem-nyomtatódó karaktereket is tartalmazzák/mutatják (4. ábra).

A 2011-es feladatsorban a minta alapján (4. ábra, fent) a hallgatóknak kellett megnevezni a hibákat. A hallgatók azonban olyan alacsony százalékban tudták csak megoldani a feladatot, hogy ezzel a módszerrel értékelhetetlen volt a teljesítményük. Ennek következtében 2012-ben változtattunk a számonkérés módján. A hibák megnevezése helyett egy egyszerűbb, ám kevésbé megbízható módszert választottunk. Az ábrát kiegészítettük a hibák sorszámozásával (4. ábra, lent), ezentúl megadtunk egy táblázatot, amelyben felsoroltuk a hibatípusokat (5. ábra, bal). A hallgatóknak az volt a feladata, hogy megjelöljék, hogy az adott sorszámú hiba melyik hibatípusba tartozik. Ezzel a módszerrel azonban két probléma is van. Egyrészt véletlenszerűen is jelölhetnek meg hibatípusokat, másrészt az is előfordulhat, hogy helyesen jelölik meg a hiba típusát, de téves ismeretek alapján [11]. Ennek ellenőrzésére 2013-ben kombináltuk az előző kétévi számonkérési módot: a táblázatot kiegészítettük a hiba megnevezésével (5. ábra, lent). Ennek a módszernek az előnye, hogy biztosan kiszűrhetőek a tévhitiek, hátránya viszont, hogy tovább csökken a helyes megoldások száma.



4. ábra: Szövegszerkesztési feladatok mintaszövege. A 2011-es nem-nyomtatódó karaktereket is tartalmazó minta (fent), valamint a sorszámozott 2012-es és 2013-es verzió (lent)

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.
helyes													
tördelési hiba													
formázási hiba													
szintaktikai hiba													
szemantikai hiba													
tipográfiai hiba													

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.
hiba megnevezése													

5. ábra: A szövegszerkesztési feladat 2012-es értékelő lapja, amelyen az egyes hibák típusát kellett megjelölni (bal), valamint a 2013-as, amelyen a hibát is kértük megnevezni (jobb)

3.2. Novemberi tesztek

Mindkét novemberi felmérésen – aktuális és következő tanév novembere – a táblázatkezelői feladatokat ismételtük meg a Debreceni Egyetem PTI, MI, GI és IK szakos hallgatóival (2. táblázat) [1], [6], [8], [9], [10], [29], [30], [31], [32]. A négy csoport közül kettő a hagyományos módon tanulta a táblázatkezelő programok használatát (GI és MI), míg kettő (PTI és IK) a CAAD-alapú deep-approach metakognitív módszerrel [7], [10].

3.3. Önértékelő lap

A szeptemberi felmérések előtt a hallgatók kitöltöttek egy önértékelő lapot, amelyen meg kellett jelöljék, hogy a felsorolt témakörökben hogyan értékelik tudásukat. Az első önértékelést 2012-ben végeztük. Az önértékelő lapon a hallgatók egy 6-fokozatú Likert-skálán jelölhették meg, hogy saját megítélésük szerint mit tudnak az adott témakörben. 2013-ban százalékos formában kellett megadni ezeket az értékeket.

3.4. Érettségi eredmények

A hallgatói teszt eredményeit összehasonlítottuk a közép- és emelt szintű érettségi eredményekkel is. Azt szerettük volna látni, hogy az érettségi eredmények korrelációban vannak-e a teszt eredményeivel, valamint azt, hogy az érettségi eredmények hogyan befolyásolják a hallgatók önértékelését.

A hallgatói teljesítmények és az érettségi eredmények összehasonlíthatóságához szükségünk lett volna a három kiválasztott téma, a szövegszerkesztés, a táblázatkezelés és a programozás részeredményeire. Az érettségiben szerzett részeredmények azonban nem állnak rendelkezésünkre, ezért a szövegszerkesztési, a táblázatkezelési és a programozási érettségi eredmények becslést értékeltek. A hallgatói informatikai és az országos informatikai érettségi eredmények ismeretében [22] megbecsültük a hallgatók részeredményeit az említett és tesztelt három témakörben.

3.5. Tanári tesztek

A tanári tesztek egyrészt tartalmazzák a hallgatói feladatokat, valamint azok értékelését. Az értékelés során a tanároktól kértük a

- teszt kitöltését;
- a hallgatói teljesítmények értékelését; százalékban kifejezve, hogy véleményük szerint a hallgatók milyen teljesítményt nyújtottak,
- a feladatok fontosságát; egy 6-fokozatú Likert-skálán jelölhették meg, hogy mennyire tartják fontosnak a feladatot.

A tesztet és a kérdőívet elküldtük az ország valamennyi általános és középiskolájának. Minösszesen 134 értékelhető, ám esetenként hiányos tesztet, kérdőívet kaptunk vissza. Itt mindenképpen érdemes megjegyezni, hogy továbbra is nyitottak vagyunk a tanári felmérés megismétlésére egy nagyobb mintán.

3.6. Feladatok és azok értékelése

A feladatok kiértékelése egy többlépcsős folyamat, annak függvényében, hogy milyen részletességgel szeretnénk elvégezni az eredmények feldolgozását.

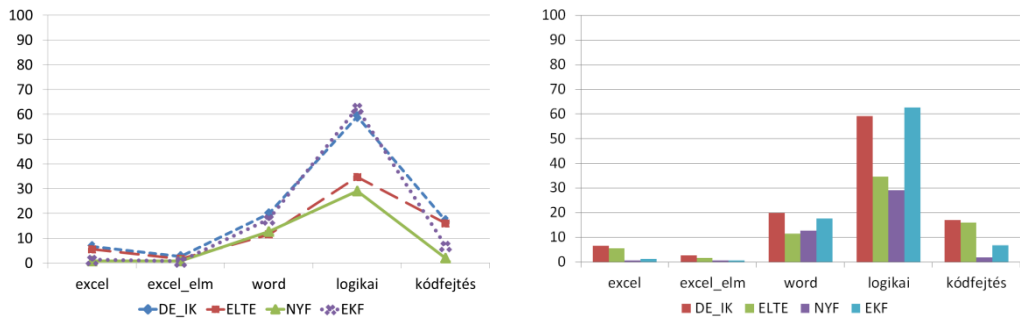
Első körben három kategóriát hoztunk létre: (1) nem válaszolt, (2) helyes, (3) nem helyes. A meg nem válaszolt kérdések, a meglepő megjegyzések és a hallgatói attitűdre vonatkozó kérdések együttese megmutatja a hallgató szakma, tanára és intézménye irányába tanúsított attitűdjét. A helyes, nem helyes válaszok megmutatják, hogy a hallgatók milyen tényleges szakmai tudással rendelkeznek, mindenféle trial-and-error wizard-based (TAEW-alapú) surface approach módszer használata nélkül [6], [8], [10]. Állítjuk ugyanis, hogy az informatikai szakembereknek, a jövő informatikusainak szüksége van arra, hogy a szakma nyelvén meg tudják fogalmazni gondolataikat. Jelen tanulmányban ezek az eredmények kerülnek bemutatásra.

A feladatok részletes értékelése alkalmas arra, hogy felfedjük, hogy milyen metakognitív döntéseket hoztak a hallgatók az egyes feladatok megoldása során [2], [10], [13], [14], [16]. Keressük a hiba okát, a hiba forrását. Ezek azok az információk, amelyek segítségével felkészíthetjük az informatikatanárokat a problémamegoldás, a hibakeresés új módszereire programozói és alkalmazói környezetben is. A papíralapú tesztelés előnye a számítógépes környezettel szemben, hogy ily módon meg tudjuk mutatni, hogy melyek azok a hibák, amelyeket a hallgatók elkövetnek. Számítógépes környezetben nem tudjuk nyomon követni a hallgatói tevékenységeket, így nem tudjuk, hogy milyen döntéssorozat eredménye a kapott dokumentum. Táblázatkezelői környezetben még inkább nehezen követhető a számítógépes tevékenység, mert a szintaktikailag helytelen képletek nem kerülnek rögzítésre, valamint arról sincs információnk, hogy TAEW- vagy CAAD-alapú döntéseket hozott a hallgató. A részletes értékeléseket és azok feldolgozását folyamatosan végezzük, melyek eredményei elérhetőek a [1], [2], [6], [8], [9], [10] publikációkban.

4. Eredmények

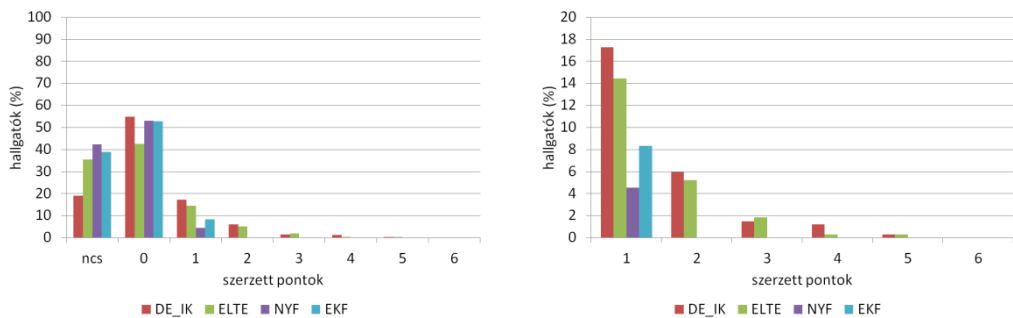
4.1. 2013-es eredmények

A 6. ábra grafikonjai egyértelműen mutatják, hogy az informatikushallgatók rendkívül alacsony teljesítményt nyújtottak. Az egyetlen feladat, amelynek elfogadható eredménye született a logikai műveletek számonkérésére adott 5–8. osztályos feladat (6. ábra, logikai). Ezt követi a szövegszerkesztési feladat (6. ábra, word) az ELTE kivételével valamennyi intézményben. A harmadik helyre került a szintén 5–8. osztályos programozási feladat megoldása (6. ábra, kódfejtés). Az ELTE-n a második helyen a programozási feladat megoldása áll. A negyedik és ötödik helyen a táblázatkezelési feladatok (6. ábra, excel) és az elméleti táblázatkezelés feladat (6. ábra, excel_elm) megoldása végzett.



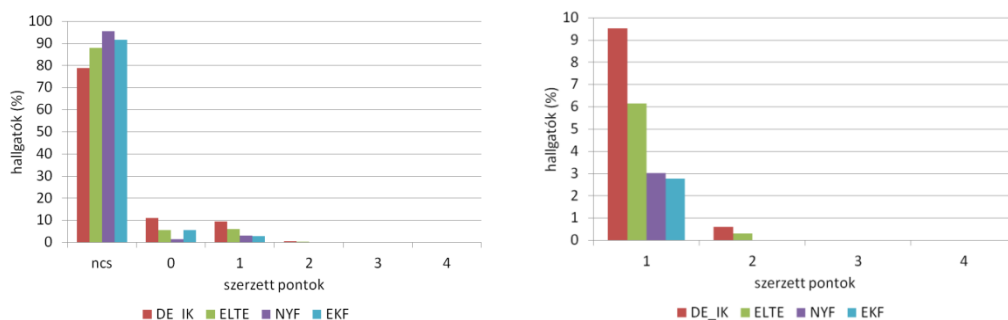
6. ábra: A 2013-as felmérés intézményenkénti átlagos eredményei

A 7–12. ábrák grafikonjai a hallgatói teljesítményeket mutatják a 2013/2014-es tanév kezdetén, nem válaszolt (ncs), helytelen (0) és az elért pontok növekvő sorrendjében. A grafikonokat végig nézve egyértelmű, hogy a hallgatók teljesítménye rendkívül alacsony, alig találni hibátlan megoldásokat. A logikai műveletek tesztelésére kitűzött általános iskolás feladat volt az, amelyet elfogadható eredménnyel oldottak meg a hallgatók (11. ábra).



7. ábra: Táblázatkezelési feladat megoldása 2013-ban

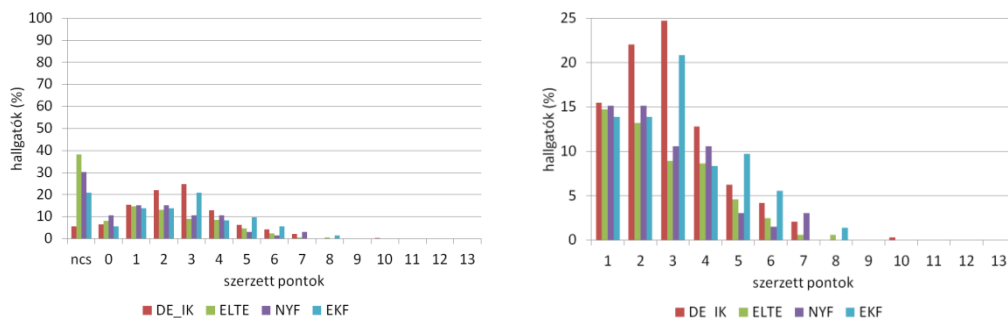
A táblázatkezelési feladatok megoldásánál egyetlen olyan hallgató sem volt a négy felsőoktatási intézményben, aki mind a hat feladatot hibátlanul meg tudta oldani, és mindösszesen kettő, aki csak egyet hibázott. Az eredmény értékelésénél mindenképpen figyelembe kell venni azt, hogy a hat feladat közül három megoldásához (2. ábra c), d) és f) feladatok) és az e) feladat megfejtéséhez ugyanaz a CAAD-alapú metakognitív döntéssorozat szükséges [7], [10]. A kapott eredmények azt mutatják, hogy a hallgatók nem fedezték fel, nem ismerték a négy feladat közötti kapcsolatot.



8. ábra: Elméleti jellegű táblázatkezelési feladat megoldása 2013-ban – az FKERES(), VKERES() függvények és az INDEX(), HOL.VAN() függvények összehasonlítása

A függvények leírása feladattal a hallgatók többsége nem foglalkozott, el sem tudta kezdeni (8. ábra, ncs). Ennek a feladatnak az eredménye a leggyengébb (6. ábra) a bemutatott öt feladat közül, alig akadt hallgató, aki egyáltalán hozzákezdett. Ahogy azt a 3.1. fejezetben említettük, azért esett a választ ezekre a függvényekre, mert a táblázatkezelési feladat első kérdése ezen függvények ismerete alapján oldható meg. A hallgatóknak fel kellett ismerni, hogy a VKERES() függvény nem használható, mert az azonos típusú adatok oszlopba vannak rendezve, az FKERES() függvény pedig azért nem, mert a találati oszlop balra helyezkedik el a keresési oszloptól, tehát az egyetlen megoldás az INDEX(HOL.VAN()) függvény alkalmazása. A függvények ismerete mindenképpen szükséges az a) táblázatkezelési feladat megoldásához. A legalább 1 pontot elért hallgatók a két feltétel egyikét tudták megfogalmazni. Nagyon kevés azoknak a száma, akik mindkét feltételt ismerték. Egyetlen hallgató sem tudta megfogalmazni, hogy a keresés csökkenő sorrendbe rendezett adatok esetén csak az INDEX(HOL.VAN()) összetett függvénnyel lehetséges, valamint azt sem, hogy az INDEX(HOL.VAN()) nemcsak egyetlen értéket, hanem egy vektort is vissza tud adni eredményként.

Sokan mondhatják és mondják is, hogy semmi szükség a függvények ismeretére, a Súlyóban úgy is le van írva minden. Ez önmagában igaz, de a felhasználók többsége nem érti a varázslók és súgók leírásait, valamint nem tanulta meg a hibaellenőrzés menetét. Ezek következménye lehet, hogy a táblázatkezelési dokumentumok több mint 90%-a hibás [24], [25], [26], [27], ami komoly anyagi- és idővesztéseget jelent a cégeknek [24], [28].



9. ábra: Szövegszerkesztési feladat megoldása 2013-ban

A szövegszerkesztési feladat értékelésénél több különböző módszerrel is próbálkoztunk (3.1. fejezet). Az eredmények azonban azt mutatják, hogy nem az értékelési rendszerrel van a prob-

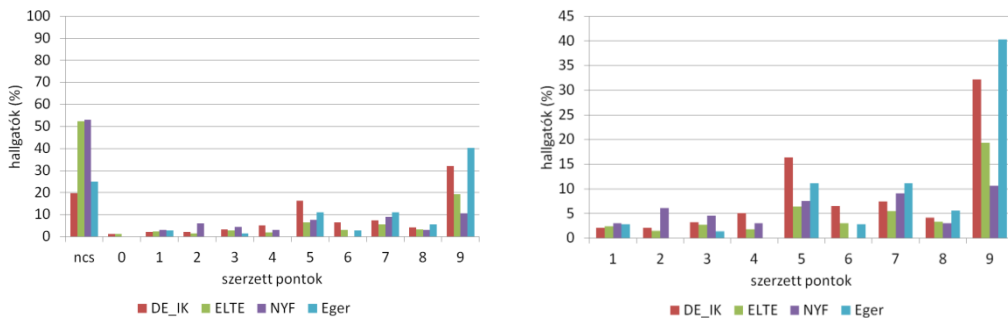
léma, hanem a szövegszerkesztési ismeretekkel. A 9. ábra grafikonjairól leolvasható, hogy a szövegszerkesztési feladatnál is nagyon alacsony teljesítményt nyújtottak a hallgatók. A 10. ábra mutatja a helyes és a még elfogadható megoldások táblázatát.

	1	2	3	4	5	6	7	8	9	10	11	12	13
helyes													
tördelési hiba	x	x		x		x	x		x		x		x
formázási hiba								x		x		x	
szintaktikai hiba					x								
szemantikai hiba													
tipográfiai hiba			x										

	1	2	3	4	5	6	7	8	9	10	11	12	13
helyes													
tördelési hiba		x		x		x	x		x		x		
formázási hiba	x		x					x		x		x	
szintaktikai hiba					x								x
szemantikai hiba													
tipográfiai hiba													

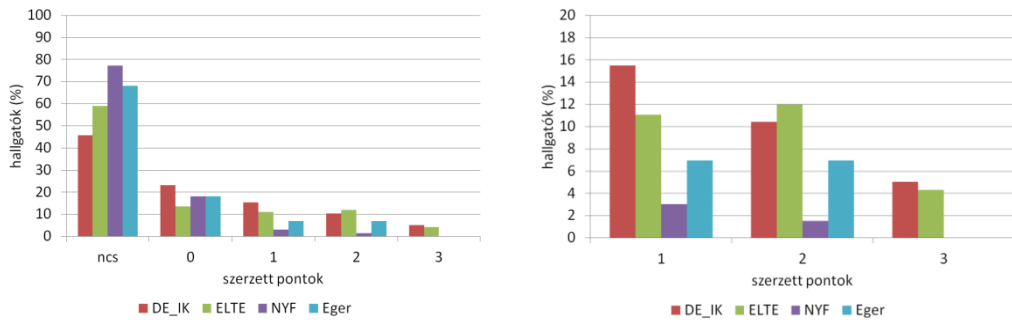
10. ábra: A szövegszerkesztési feladat lehetséges megoldásai. A bal oldali tábla mutatja a helyes megoldást, míg a jobb a még elfogadható megoldásokat

Mindhárom értékelési mód azt adta eredményül, hogy annyit tudnak a hallgatók, hogy Entert és szóközt nyomogatni nem illik szövegszerkesztési céllal. Nem minden esetben ismerik fel a problémát, de mondhatjuk, hogy ezt elfogadható biztonsággal tudják. A leggyakoribb helyesen megjelölt hibaszám a három. A hiba típusának megjelöléséből már sejthető, míg a hiba megnevezéséből biztosan tudjuk, hogy az Enter és szóközhibák adják ezt az értéket. A nyolc tördelési hibából hét tartalmaz Enter és szóköz hibát (4. és 10. ábra, 2., 4., 6., 7., 9., 11., és 13. hibák). Ez azt jelenti, hogy a lehetséges hétből mindösszesen három ilyen jellegű hibát ismernek fel a hallgatók nagy valószínűséggel. Ez rendszerint a 2. a 11. és a 6. vagy 7. hiba. Mindenképpen érdemes megjegyezni, hogy a hasonló 6. és 7. hibák felismerése nem mutatott korrelációt [11].



11. ábra: A logikai műveletek eredménye 2013-ban

A 11. ábra mutatja a logikai műveletek tesztelésére adott feladat eredményét. Az eredményeket tartalmazó grafikonokat összehasonlítva egyértelmű, hogy ennek a feladatnak az eredményei a legjobbak (6. ábra). Öröndetes, hogy ennél a feladatnál igen magas számban találtunk hibátlan megoldásokat. Mindenképpen érdemes azonban megjegyezni, hogy a tanárok véleménye szerint a hallgatók gyengébb teljesítményt nyújtanak, mint szövegszerkesztésben (18. ábra).



12. ábra: Programozási feladat megoldása 2013-ban

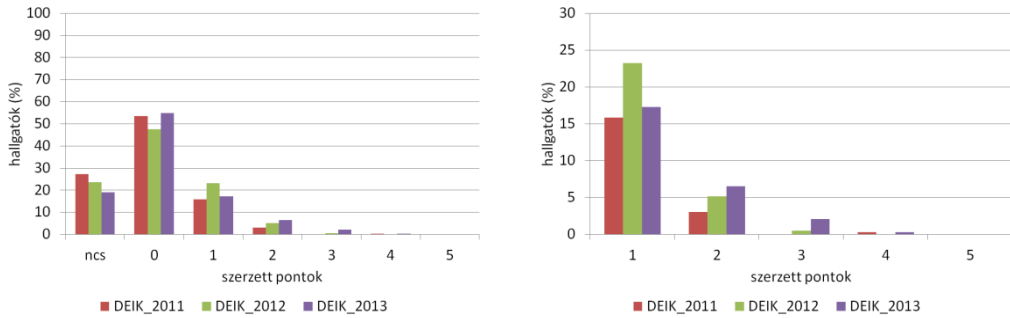
A 12. ábráról leolvasható, hogy nagyon magas azoknak a hallgatóknak a száma, akik hozzá se kezdtek a programozási feladatok megoldásához. Még a legjobban teljesítő Debreceni Egyetemen is a hallgatók közel 50% meg se próbálta a megoldani a feladatot. Figyelembe véve, hogy a mindhárom pseudo kód egy 5–8. osztályos feladatsorból vett részlet – tehát viszonylag egyszerű, alapfeladat – az eredmények nagyon alacsonyok. A feladat egyszerűségén túl azt sem szabad elfelejtenünk, hogy informatikushallgatók töltötték ki a tesztet, akiknek a programozás lesz a szakmájuk. A feladat megoldása azt mutatja, hogy a hallgatók alapvető algoritmizáló készséggel sem rendelkeznek legalább 6–8 év informatikatanulás után. Még azon hallgatók közül is nagyon kevesen tudták hibátlanul megoldani a feladatokat, akik emelt szintű érettségít tettek. A feladatok részletes értékeléséhez a SOLO-taxonómia kategóriáit használtuk [12]. Az eredményeket a [2] publikáció tartalmazza. A SOLO-taxonómia a megértésnek az alábbi 5 kategóriáját különbözteti meg [12], [14]:

- struktúra nélküli (Pre-structural)
- egyszerű strukturális (Uni-structural)
- multi-strukturális (Multi-structural)
- relációs (Relational)
- kiterjesztett absztrakt (Extended abstract)

A struktúra nélküli és kiterjesztett absztrakt megértési szintek a feladatok jellegéből adódóan nem értékelhetők. Az egyszerű-, a multi-strukturális és a relációs megértési szintek alapján értékeltük a hallgatói teljesítményeket.

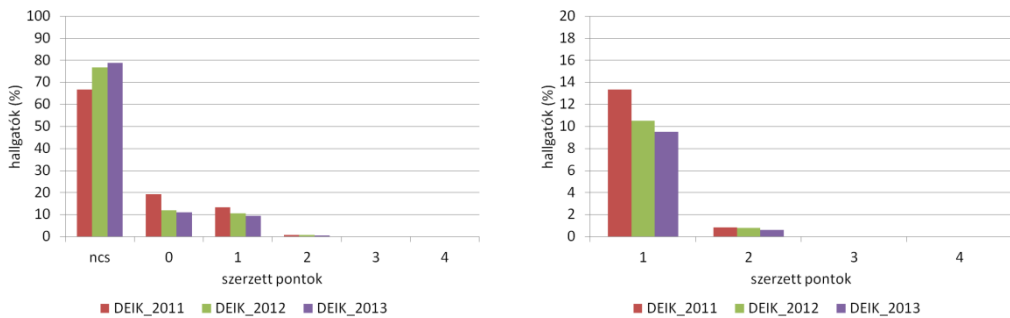
4.2. Debreceni Egyetem hallgatóinak eredményei

A Debreceni Egyetem Informatikai Karának hallgatóit a 2011/2012-es tanévtől kezdődően mindhárom évben teszteltük. A felmérésben négy csoport minden évben részt vett, a kar PTI, MI, GI és IK szakos hallgatói. A következőkben a 13–17. ábrák grafikonjaival bemutatjuk DE IK hallgatók teljesítményének alakulását.



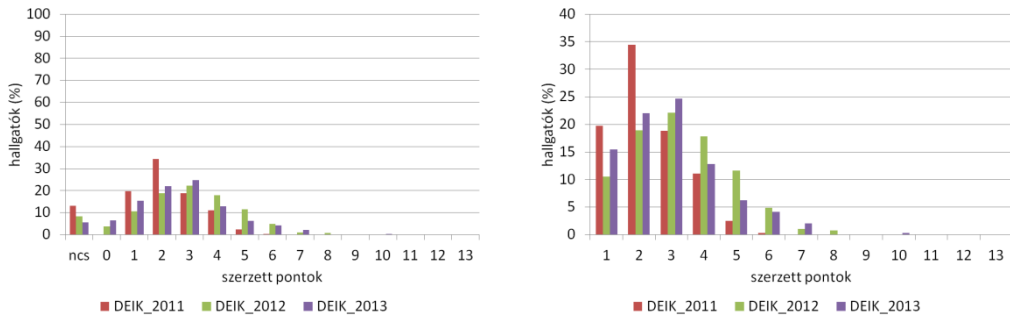
13. ábra: DE IK hallgatók táblázatkezelői feladatainak eredményei

A három év tesztelése során azt tapasztaltuk, hogy csökken azoknak a hallgatóknak a száma, akik nem foglalkoznak a táblázatkezelési feladattal. A teljesen helytelen megoldások számában nincs lényeges változás. A 2012-es csökkenés azt eredményezte, hogy emelkedett az egy kérdést helyesen megválaszolók száma. Folyamatos emelkedés tapasztalható a kettő és három helyes választ adók számában.



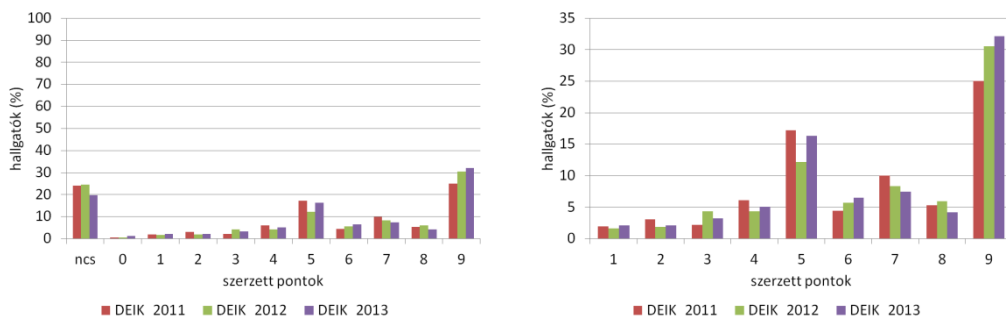
14. ábra: DE IK hallgatók elméleti jellegű táblázatkezelői feladatának eredményei

Ellentétben a gyakorlati táblázatkezelői feladattal, az elméleti feladatnál folyamatosan nőtt azon hallgatók száma, akik nem foglalkoztak a feladattal, meg se próbálták. Ezzel párhuzamosan csökkent a helyes megoldást adók száma. Egyetlen olyan hallgatót sem találtunk, aki maradéktalanul le tudta írni a függvények tulajdonságait, alkalmazhatóságát. Már a két feltételt megfogalmazók száma is rendkívül alacsony.



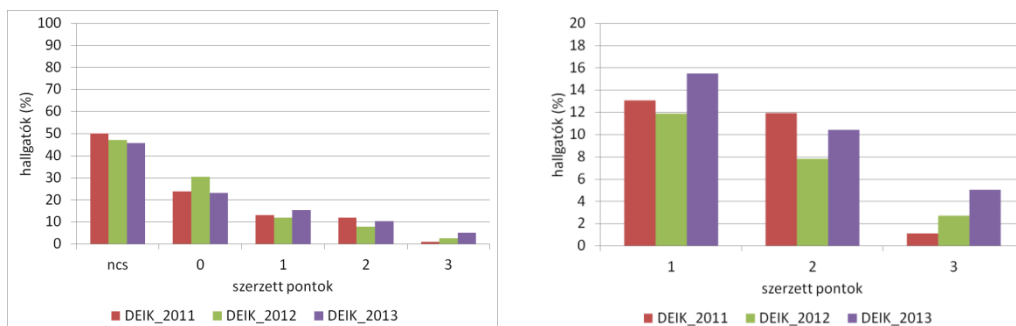
15. ábra: DE IK hallgatók szövegszerkesztési feladatainak megoldása

A szövegszerkesztési feladat megoldásánál a lehetséges 13 pontból 2 pont volt a leggyakoribb 2011-ben, és 3 pont 2012-ben és 2013-ban. Ahogy a 2013-as eredményeknél említésre került (4.1. fejezet), az Enterrel és szóközzel szerkesztést történő képesek felismerni és hibás-ként megjelölni a hallgatók. Az viszont egyértelmű, hogy a hallgatók nem ismerik a helyesen szerkesztett dokumentum fogalmát [3].



16. ábra: DE IK hallgatók logikai műveletek feladatainak megoldása

A logikai műveletek tesztelésére adott feladat megoldása során a hibátlan megoldások száma évről évre emelkedik. Hasonlóan az országos eredményekhez, 5 pont a második leggyakoribb eredmény. Ennek magyarázata, hogy a hallgatók a feladat 6–8. kérdéseit válaszolták meg hibásan, ugyanazt a hibát elkövetve.

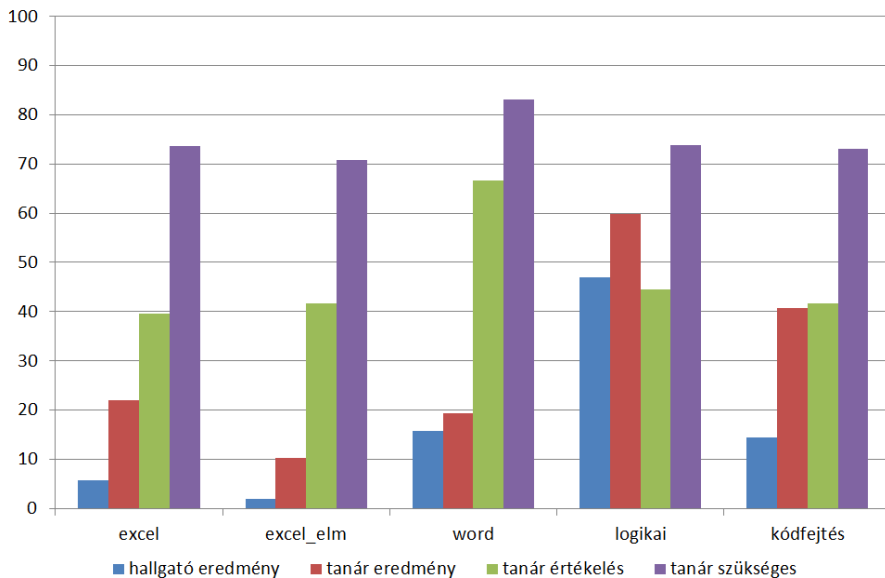


17. ábra: DE IK hallgatók programozás feladatának megoldása

A programozás feladatnál is igaz, hogy nagyon magas azoknak a száma, akik nem foglalkoztak a feladattal és alacsony a helyes válaszok száma. Enyhe csökkenés tapasztalható azonban a nem foglalkozott vele és a 0 pontos megoldások számában, és egy enyhe növekedés a helyesen megoldott feladatok számában. Különösen örvendetes, hogy nőtt azoknak a száma, akik mindhárom feladatot meg tudták oldani.

4.3. Érettségi eredmények, tanári- és önértékelés

Összehasonlítva a teszt eredményeit (1) a hallgatók érettségi eredményével, (2) a tanárok teszt eredményével, (3) a várható hallgatói teljesítmények és (4) a feladatok szükségességének tanári értékelésével, valamint (5) a hallgatók önértékelésével választ kaphatunk az előzőekben megfogalmazott kérdések némelyikére.



18. ábra: A hallgatói, a tanári teljesítmények, a hallgatói teljesítmények tanári értékelése, a feladatok szükségességének tanári értékelése

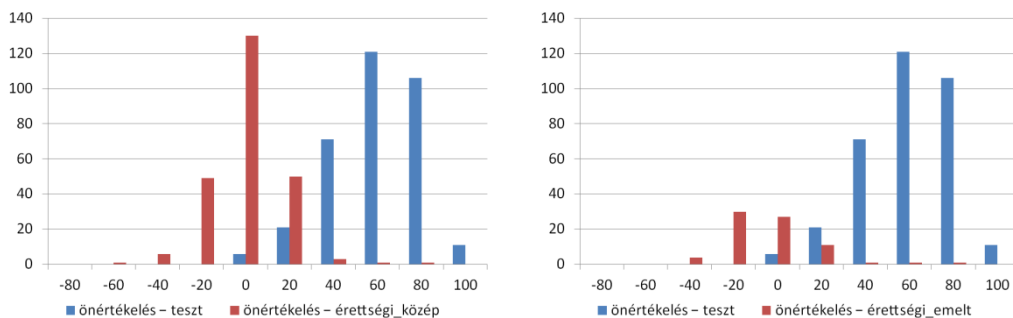
A 18. ábra összehasonlító adataiból a következők olvashatók le:

- A tanárok véleménye szerint valamennyi feladat fontos, több mint 70%-ra értékelték a feladatok szükségességét (70,83–83,18%). A szövegszerkesztés feladatot gondolták a legfontosabbnak, a táblázatkezelési elméleti feladatot a legkevésbé fontosnak, a többi feladat közel azonos értéket kapott.
- A logikai feladatot kivéve a tanárok minden esetben olyan magas értékre gondolták a hallgatói teljesítményt, ami bőven meghaladta nemcsak a hallgatói teljesítményeket, hanem a saját teljesítményüket is.
- A tanárok a szövegszerkesztési feladat eredményét gondolták a legjobbnak (66,71%), a többi feladatot közel azonos eredményt adott. Ezek közül a táblázatkezelési gyakorlati feladat végzett az utolsó helyen (39,55–44,54%). Úgy gondolták, hogy az összes feladattípus közül a táblázatkezelést tudják a legkevésbé. Ezen a ponton érdemes megemlíteni, hogy a tanárok 11,94% úgy ítélte meg és válaszként azt adta, hogy a megfejtendő táblázatkezelési feladat hibás (2. ábra, e) feladat).
- A tanárok véleménye szerint a táblázatkezelési feladatoknál a gyakorlat fontosabb, mint az elmélet (excel=73,65%, excel_elm=70,83%), de ennek ellenére úgy vélik, hogy az elméleti feladatot oldják meg jobban a hallgatók (excel=39,55%, excel_elm=41,75%).

Az informatikushallgatók alacsony teljesítménye egy rendkívül összetett probléma. Számos magyarázata lehet annak, hogy miért is alakult így az informatikaoktatásunk. Az okok és magyarázatok keresése érdekében hasonlítottuk össze a hallgatói és a tanári teljesítményeket, valamint a tanári értékeléseket. A 18. ábra grafikonjáról egyértelműen leolvasható információ mindenképpen kérdéseket vet fel. Ezek közül sorolunk fel néhányat, de hangsúlyoznunk kell, hogy az itt feltett kérdések csak egy szűk válogatás abból, amelyek a kapott eredmények és eredménytelen- ségek láttán megfogalmazódnak.

- Miért ilyen alacsonyak a tanári teljesítmények?

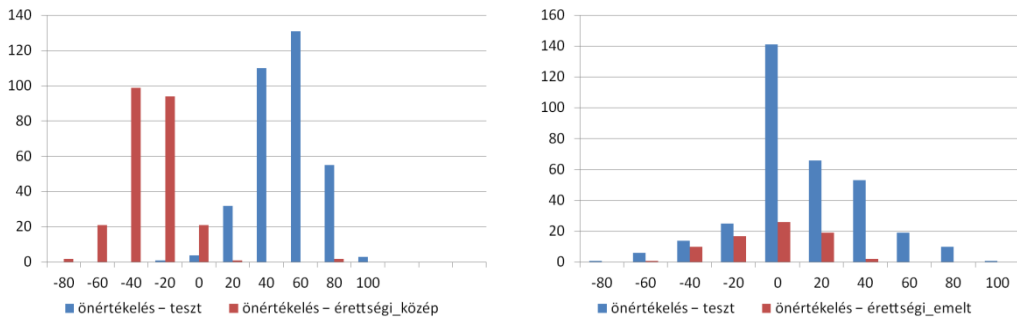
- Miért nem tudják a tanárok megbecsülni a hallgatói teljesítményeket?
- Miért van az, hogy egyes feladatoknál a tanárok alig tudnak többet, mint a hallgatók?
- Miért van az, hogy a tanárok nem ismerik a függvényeket? Vajon hogyan tanítják meg? Mit mondanak, hogy mikor melyiket lehet és érdemes használni?
- Miért nem tudnak egy általános iskolás pseudo kódot megfejteni és magyarul megfogalmazni a választ?
- Mit jelent számukra a szövegszerkesztés, ha nem tudják eldönteni, hogy egy dokumentum helyes vagy sem?



19. ábra: Táblázatkezelési gyakorlati feladat eredményének összehasonlítása a becsült érettségi eredményekkel közép- (bal) és emelt szinten (jobb)

A tanári értékeléseken túl egy másik mérője a hallgatói tevékenységeknek az érettségi. A 19. és 20. ábrák a DE IK hallgatók informatikai érettségi eredményei alapján készültek, de a többi intézmény esetében is hasonló eredményeket kaptunk, amit helyhiány miatt jelen tanulmányban nem közlünk részletesen [9]. Összehasonlítottuk az önértékelési és a teszt eredményeket, valamint az önértékelési és az érettségi eredményeket, majd a kiválasztott két érték különbségét számoltuk ki minden egyes hallgató esetében. Az így kapott értékek gyakoriságát ábrázoltuk a 19. és 20. ábrákon.

A táblázatkezelési feladatok esetén a 19. ábra grafikonjai egyértelműen mutatják, hogy lényegesen magasabbra értékelték a tudásukat a hallgatók, mint a teszt eredmények. A középszintű érettségi eredményekkel összehasonlítva az önértékelést azt látjuk, hogy a hallgatók többsége úgy gondolja, hogy az érettségi eredménye jobb, mint a tényleges tudása (19. ábra, bal). Mindezt összevetve azt a következtetést vonhatjuk le, hogy az informatikushallgatók szerint a középszintű érettségi nem méri a tényleges tudásukat, hanem egy a tényleges tudásnál magasabb értéket ad. Az emelt szintű érettségi eredményeit összevetve az önértékeléssel, már egy sokkal jobb képet kapunk: a hallgatók úgy gondolják, hogy az emelt szintű érettségi, megközelítőleg mutatja a tényleges tudásukat (19. ábra, jobb).



20. ábra: A szövegszerkesztési (bal) és a programozási, kódfejtesi (jobb) feladat eredményének összehasonlítása a becsült közép- (bal) és emelt szintű (jobb) érettségi eredményekkel

A szövegszerkesztési feladat megoldása és a szövegszerkesztés önmagában is problémásnak tűnik a felmérés alapján. A tanárok úgy gondolják, hogy nagyon fontos (18. ábra), úgy gondolják, hogy a hallgatók nagyon tudják (18. ábra), ugyanakkor az adatok azt mutatják, hogy sem a tanárok, sem a hallgatók nem tudják (9., 15. és 18. ábra). Az ismereteik addig terjednek, hogy Entert és szökőzt ne használjunk szövegszerkesztéshez. A hallgatók nagy többsége lényegesen magasabbra értékeli szövegszerkesztési tudását, mint azt a felmérés adatai mutatják (20. ábra, bal). A szövegszerkesztésnél még inkább jellemző, hogy a középszintű érettségi eredményei lényegesen magasabbak, mint a hallgatói önértékelés. Ebben az esetben tehát még inkább elmondható, hogy a középszintű érettségi nem a szövegszerkesztési ismereteket méri, hanem sokkal inkább azt, hogy milyen TAEW-alapú metakognitív döntéssorozatot képes a vizsgázó végrehajtani. A minta és az utasítássorozat alapján történő tesztelés nem várja el a vizsgázótól CAAD-alapú metakognitív döntések meghozatalát.

A programozási feladatok esetében a hallgatói önértékelések és a teszt eredményei alapján azt kaptuk, hogy a hallgatók viszonylag jól meg tudják ítélni, hogy meg tudnak-e oldani egy 5–8. osztályos feladatot. Ebben az esetben is tapasztalható egy túlzott önértékelés, de nem annyira erős, mint a többi feladatnál. Nagyon fontos azonban hangsúlyoznunk, hogy míg a szövegszerkesztési és táblázatkezelési feladatok az életkori sajátosságoknak megfelelő feladatok voltak, addig a programozás feladat egy általános iskolás feladat. Az önértékelést összehasonlítva az emelt szintű érettségi eredményekkel azt tapasztaljuk, hogy a programozás esetében áll a két érték legközelebb egymáshoz. A hallgatók tehát úgy gondolják, hogy az emelt szintű programozási eredmények jól mutatják a programozás tudásukat, tehát azt, hogy rendelkeznek-e egy 5–8. osztályostól elvárt algoritmikus készséggel.

A programozási feladatok részeredményeinek az értékelése során azt tapasztaltuk, hogy a csak középszintű érettséggel rendelkező hallgatók algoritmikus készsége a legalacsonyabb, összehasonlítva azokkal, akik emelt szintű érettségit tettek és azokkal, akik érettségi vizsgájáról nincsen információnk [2]. Ez az eredmény, eredménytelenség azt mutatja, hogy az alkalmazói ismeretek tanítása nem algoritmikus szemlélettel történik, ennek következtében nem fejlődik a hallgatók algoritmikus készsége, ami meggátolja őket a hatékony számítógépes problémamegoldásban.

4.4. A novemberi tesztek

Tapasztalataink azt mutatják, hogy az informatikai nemtudásnak az egyik oka, hogy a hallgatók TAEW-alapú metakognitív lépéssorozatokat hajtanak végre, amely során elvész az algoritmikus gondolkodás, és a felhasználók mindféle háttértudás nélkül használják az eszközöket [2]. Ennek

ellensúlyozására bevezettünk egy CAAD-alapú deep-approach jellegű módszert a táblázatkezelés oktatásához [6], [7]. A módszer lényege, hogy a lehető legkevesebb számú egyszerű függvényt tanítsunk csak meg, valamint azt, hogy hogyan lehet ezeket többszintű függvényekké összeépíteni. Tehát nem tettünk mást, mint azt, hogy a táblázatkezelőt használtuk első programozási nyelvként [33]. Ennek a módszernek a tesztelését végeztük a novemberi felmérések során. A módszer részletes leírását, az előnyeit és a hatékonyságvizsgálat eredményeit a [1], [6], [8], [9], [10] publikációkban ismertettük.

A CAAD-alapú módszer bevezetése és eredményei egyértelműen mutatják, hogy az alkalmazói szoftverek is taníthatók algoritmikus szemlélettel, valamint azt, hogy az így megszerzett tudás megbízhatóbb és tartósabb, mint a felületkezelés során, TAEW-alapú módszerekkel szerzett ismeretek.

5. Összegzés

Az év eleji felmérések eredményei egyértelműen mutatják, hogy az Informatikai Karon tanulmányaikat megkezdő hallgatók informatikai ismeretei rendkívül hiányosak. A kitűzött programozási feladatokat a hallgatók 3,88–25,91% oldotta meg hibátlanul 2013-ban, és egyetlen olyan hallgató és tanár sem volt, aki az alkalmazói ismereteket számon kérő feladatokat meg tudta oldani hibátlanul. Az előző évek felmérései is hasonló eredményeket mutattak.

Meg lehet magyarázni és továbbra is lehet kifogásokat keresni, hogy az alkalmazói ismereteket számon kérő feladatok eredménye azért ilyen alacsony, mert azokat számítógépes környezetben kell megoldani. Sajnos, a felmérések azt mutatják, hogy az alkalmazói szoftverekkel készített dokumentumok is megközelítően ilyen százalékban tartalmazznak hibákat, komoly anyagi veszteségeket okozva a cégeknek [24], [25], [26], [27], [28]. Valami egészen más lehet az alacsony eredmény oka, nem a számítógép hiánya a tesztelés során. Ezeket az okokat kell megtaláljuk és olyan módszereket bevezetni, amelyekkel javítható a tanári és a hallgatói teljesítmény is.

A feldolgozás következő fázisaiban a hallgatói teljesítményeket összehasonlítottuk a hallgatók érettségi eredményével és az országos érettségi átlagokkal, a hallgatói önértékelésekkel és a tanári értékelésekkel.

A hallgatói teljesítmények, az érettségi eredmények, a hallgatói önértékelések összehasonlítása a következő eredményekhez vezetett:

- A hallgatók papíralapú teljesítménye lényegesen elmarad az érettségi eredményektől.
- A hallgatók úgy ítélik meg, hogy az érettségi eredmények nem tükrözik a tényleges tudásukat, a tényleges tudásuk alacsonyabb, mint ahogy azt az érettségi eredmények mutatják.
- A hallgatók felülértékelik tudásukat.
- A hallgatók szóbeli kifejező készsége, terminológiai ismerete lényegesen alulmarad a feladatmegoldó készségüktől. Jelen tanulmányban csak egyetlen ilyen példát mutattunk, de a teszt további elméleti jellegű feladatai hasonló alacsony eredményeket mutatnak.
- Ez utóbbi megállapítás ellentmond az érettségi gyakorlati és szóbeli eredményeinek, amelyek egyértelműen azt mutatják, hogy a hallgatók jobban teljesítenek szóban, ha beszélni kell a szakmáról, mintha számítógép mellett kell a problémát megoldani.
- Az érettségi eredmények ellentmondásosak, nem tükrözik a tényleges tudást.
- A tanárok nem tudják megítélni a hallgatói teljesítményeket, a hallgatók tudását.

- A csak középszintű érettségit letett hallgatók nem vagy csak rendkívül alacsony szintű algoritmikus készséggel rendelkeznek [2].
- Az előző felismerés súlyos következménye, hogy a hallgatók többsége az informatikában nem látja az algoritmikus megközelítés lehetőségét, csak alkalmazói szoftvereket használ és ezeken is csak felületkezelést végez, amely tevékenységek a TAEW-alapú surface-approach metakognitív megközelítések csoportjába tartoznak.

A hallgatói alacsony tudásszintnek számtalan magyarázata van és lehet. Ezek részletes kifejtése természetesen meghaladja jelen tanulmány kereteit. Az azonban mindenképpen továbbgondolásra kell, hogy készítsen bennünket, hogy az informatikatanárok nem tudják, hogy tanítványaik mit tudnak, valamint az, hogy nem algoritmikus szemlélettel tanítjuk az informatikát, hanem csak a kattintgatást. Sokan azzal takaróznak, hogy nincs szükségük az algoritmikus készség kialakítására, mivel nem fognak programot írni. Ez azonban nem elfogadható magyarázat. Az informatikának, a számítástechnikának a surface-approach típusú metakognitív megközelítése vezet olyan dokumentumok létrejöttéhez, amelyek háttérismeretek nélkül, ennek következtében ellenőrizetlenül, ennek következtében hibásan kerülnek mentésre, publikálásra. Tovább nehezíti az alkalmazói szoftverek helyes használatát, hogy jelenleg nem vagy csak rendkívül speciális esetekben állnak rendelkezésünkre olyan módszerek, amelyek alkalmasak a különböző típusú felhasználói dokumentumok tesztelésére. Ezen módszerek hiányában a felhasználó feladata lenne a dokumentumok helyességének ellenőrzése, amelyhez több tudományterület ismereteinek együttes jelenléte szükséges [11].

A felmérés eredménye egyértelműen mutatja, hogy az informatikushallgatók többsége az egyetemi tanulmányaik megkezdésekor nem rendelkezik megfelelő szintű algoritmikus készséggel, szakmai és terminológiai ismerettel. Ezen készségek fejlesztésére az informatikatanárokat fel kell készíteni, a tanárképzést át kell alakítanunk. Olyan tanárokat kell képeznünk, akik ezt meg tudják célként fogalmazni a különböző szintű tanterveinkben is, különös figyelemmel a Nemzeti Alaptantervre és a kerettantervre, valamint olyanokra, akik ezt meg is tudják tanítani.

Köszönetnyilvánítás

A publikáció elkészítését a TÁMOP-4.2.2.C-11/1/KONV-2012-001 számú projekt támogatta. A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

A kutatást részben támogatta az OTKA K-105262 számú projekt.

Irodalom

1. Biró Piroska, Csernoch Mária: *Táblázatkezelés algoritmikus megközelítése*. Kiss Árpád Emlékkonferencia Tanulmánykötete, Debrecen, (2013). Elfogadva.
2. Biró Piroska, Csernoch Mária: *Deep and surface structural metacognitive abilities of the first year students of Informatics*. 4th IEEE International Conference on Cognitive Infocommunicaitons, Budapest (2013). Elfogadva.
3. Biró Piroska, Csernoch Mária: *Elsőéves informatikushallgatók algoritmizáló készségei*. XXIII. Nemzetközi Számítástechnika és Oktatás Konferencia 2013. EMT Kiadó, (2013), 154-159 old.
4. Csernoch Mária: *Teaching word processing – the theory behind*. Teaching Mathematics and Computer Science. 2009/1.
5. Csernoch Mária: *Teaching word processing – the practice*. Teaching Mathematics and Computer Science. 8/2 (2010) pp.247–262.

6. Csernoch Mária: *Introducing Conditional Array Formulas in Spreadsheet Classes*. EDULEARN12 Proceedings. Barcelona, Spain. 2-4 July, 2012. Publisher: IATED, (2012) 7270–7279.
7. Csernoch Mária, Balogh László: *Algoritmikus és táblázatkezelés. Tehetséggondozás a közoktatásban az informatika terén*. Magyar Tehetségszolgáltató Szervezetek Szövetsége, Budapest. (2011) ISSN 2062-5936.
8. Csernoch Mária, Biró Piroska: *Button-up technikák hatékonyságának vizsgálata informatika szakos hallgatók táblázatkezelés-oktatásában*. Szerk: Kozma Tamás és Perjés István, Új kutatások a neveléstudományokban 2012, ELTE Eötvös Kiadó, (2013) 369–392.
9. Csernoch Mária, Biró Piroska: *Teachers' Assessment and Students' Self-Assessment on The Students' Spreadsheet Knowledge*. EDULEARN13 Proceedings July 1st-3rd, 2013 — Barcelona, Spain. Publisher: IATED, (2013) 949–956.
10. Csernoch Mária, Biró Piroska: *Spreadsheet misconceptions, spreadsheet errors*. Hungarian Conference on Educational Research, Debrecen, (2013). Elfogadva.
11. Csernoch Mária, Biró Piroska: *Az algoritmikus szemlélet hiánya és ennek következményei a szövegszerkesztés-oktatásban*. XIII. Országos Neveléstudományi Konferencia. Eszterházy Károly Főiskola Eger 2013. november 7-8-9.
12. Biggs J. B., Collis K. E.: *Evaluating the Quality of Learning: The SOLO Taxonomy*. New York: Academic Press, (1982).
13. Csikos Csaba: *Metakogníció*. A tudásra vonatkozó tudás pedagógiája. Műszaki Kiadó. Budapest, (2006).
14. Csapó Benő és Szabó Gábor (szerk.): *Tartalmi keretek a természettudomány diagnosztikus értékeléséhez*. Nemzeti Tankönyvkiadó, (2012), Budapest. 9-15.
15. Case, J. M., Gunstone, R. F.: *Approaches to learning in a second year chemical engineering course*. International Journal of Science Education, (2003) vol. 25 (7), pp. 801–819.
16. Case, J. M., Gunstone, R. F.: *Metacognitive development as a shift in approach to learning: an in-depth study*. Studies in Higher Education, (2002) vol. 27(4), pp. 459-470.
17. Justin Kruger, David Dunning: *Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments*. Journal of Personality and Social Psychology (1999) Vol. 77, No. 6, 1121-1134.
18. *Kerettanterv* (2013). <http://kerettanterv.ofi.hu/index.html>. Letöltés dátuma: 2013.08.30.
19. *Kerettanterv* (2008). Magyar Közlöny, 20. szám II. kötet, 2008. február 8.
20. *Nemes Tihamér Országos Informatikai Verseny 2008-2009* I. forduló, I. korcsoport, 5-8. osztály. Feladatok. <http://nemes.inf.elte.hu/2009/nt09-1fl.doc>. Letöltés dátuma: 2013.08.30.
21. *Nemes Tihamér Országos Informatikai Verseny 2008-2009* I. forduló, I. korcsoport, 5-8. osztály. Megoldási és értékelési útmutató. <http://nemes.inf.elte.hu/2009/nt09-1ml.doc>. Letöltés dátuma: 2013.08.30.
22. *Érettségi eredmények 2007-2013*. <https://www.ketszintu.hu/publicstat.php>. Letöltés dátuma: 2013.08.30.
23. *Excel functions (by category)*. *Applies to Excel 2010*. <http://office.microsoft.com/en-us/excel-help/excel-functions-by-category-HP010342656.aspx>. Letöltés dátuma: 2013.10.12.
24. Panko, R. R.: *What We Know About Spreadsheet Errors*. Journal of End User Computing's. Special issue on Scaling Up End User Development. (2008) (10)2, 15-21.
25. Powell, S. G., Baker, K. R., Lawson, B.: *A critical review of the literature on spreadsheet errors*. Decision Support Systems, (2008) 46(1), 128–138.
26. Powell, S. G., Baker, K. R., Lawson, B.: *Errors in operational spreadsheets*. Journal of Organizational and End-User Computing, (2009) 1(3), 4–36.
27. Powell, S. G., Baker, K. R., Lawson, B.: *Impact of errors in operational spreadsheets*. Decision Support Systems, (2009) 47(2), 126–132.

28. *Az olasz, magyar, görög után ezúttal egy holland tanulmány a tudatlanság áráról.* Az ECDL Alapítvány március 9-i sajtóközleménye alapján. Mi újság. 2012. április.
http://njszt.hu/sites/default/files/mi_ujsg_2012_aprilis.pdf
29. Wakeling, D.: *Spreadsheet functional programming*. JFP 17(1), 131–143, 2007. Cambridge University Press, (2007).
30. Walkenbach, J.: *Excel2003 Formulas*. John Wiley & Sons, (2003).
31. Walkenbach, J., Wilcox, C.: *Putting basic array formulas to work*, (2003).
<http://office.microsoft.com/en-us/excel-help/putting-basic-array-formulas-to-work-HA001087292.aspx?CTT=5&origin=HA001087290>. Retrieved May 8, (2012).
32. Wilcox, C., Walkenbach, J.: *Introducing array formulas in Excel*, 2003. <http://office.microsoft.com/en-us/excel-help/introducing-array-formulas-in-excel-HA001087290.aspx>. Retrieved April 12, (2013).
33. Warren, P.: *Learning to program: spreadsheets, scripting and HCI*, in Proceedings of the Sixth Australasian Conference on Computing Education - Volume 30, Darlinghurst, Australia, (2004), 327–333.