



**Séma- és tudástranszferalapú számítógépes problémameg-  
oldási képességek fejlesztése az informatikai alkalmazói ismeretek tanítása során**

Egyetemi doktori (PhD) értekezés

a szerző neve: Sebestyén Katalin

témavezető neve: Dr. Csernoch Mária

DEBRECENI EGYETEM  
Természettudományi és Informatikai Doktori Tanács  
Informatikai Tudományok Doktori Iskola  
Debrecen, 2021.

Ezen értekezést a Debreceni Egyetem Természettudományi Doktori Tanács Informatikai Tudományok Doktori Iskola Alkalmazott információ technológia és elméleti hátttere programja keretében készítettem a Debreceni Egyetem műszaki doktori (PhD) fokozatának elnyerése céljából.

Nyilatkozom arról, hogy a tézisekben leírt eredmények nem képezik más PhD disszertáció részét.

Debrecen, 2021.

.....  
Sebestyén Katalin

Tanúsítom, hogy Sebestyén Katalin doktorjelölt 2017 - 2021 között a fent megnevezett Doktori Iskola Alkalmazott információ technológia és elméleti hátttere programjának keretében irányítással végezte munkáját. Az értekezésben foglalt eredményekhez a jelölt önálló alkotó tevékenységével meghatározóan hozzájárult. Nyilatkozom továbbá arról, hogy a tézisekben leírt eredmények nem képezik más PhD disszertáció részét.

Az értekezés elfogadását javaslom.

Debrecen, 2021.

.....  
Csernoch Mária

**SÉMA- ÉS TUDÁSTRANSZFERALAPÚ SZÁMÍTÓGÉ-  
PES PROBLÉMAMEGOLDÁSI KÉPESSÉGEK FEJLESZ-  
TÉSE AZ INFORMATIKAI ALKALMAZÓI ISMERETEK  
TANÍTÁSA SORÁN**

Értekezés a doktori (PhD.) fokozat megszerzése érdekében

Az Informatika didaktika tudományágban  
Írta: Sebestyén Katalin okleveles informatikatanár – könyvtárpedagógia-tanár

Készült a Debreceni Egyetem Informatikai Tudományok doktori iskolája  
Alkalmazott információ technológia és elméleti háttere programja keretében

Témavezető: Dr. Csernoch Mária

Az értekezés bírálói:

Bujdosóné Dr. Dani Erzsébet .....  
Dr. Stoffa Veronika .....

A bírálóbizottság:

elnök: Dr.....  
tagok: Dr.....  
Dr.....  
Dr.....  
Dr.....

Az értekezés védésének időpontja: .....

## Tartalomjegyzék

1. Bevezetés.....	1
1.1. Célkitűzés.....	2
2. Tézisek.....	3
3. Informatikaoktatás helyzete Magyarországon.....	6
3.1. Tartalmi szabályozók.....	6
3.1.1. Alkalmazói ismeretek fontossága és a NAT.....	9
3.2. PISA és a digitális szövegértés.....	10
3.3. Kihívások, nehézségek a pedagógusok számára.....	12
4. Előzetes vizsgálatok és eredmények.....	14
4.1. Algoritmikus és alkalmazási készségek tesztelése – TAaAS.....	14
5. Módszerek.....	20
5.1. Problémamegoldás.....	21
5.2. Sémaépítés.....	23
5.3. Didaktika és a problémamegoldás.....	25
6. Kutatócsoport által fejlesztett és alkalmazott módszerek.....	27
6.1. Fájlkezelés.....	28
6.1.1. Fájlonverzió (WDC).....	37
6.2. Szövegkezelés.....	39
6.2.1. Hibafelismerés és keresés (ERM).....	43
6.3. Táblázatkezelés.....	52
6.3.1. Spreadsheet Lego (Sprego).....	54
6.4. Oktatási segédeszközök.....	58
6.4.1. 2D oktatóprogram.....	60
6.5. Szerepvállalás a kutatócsoportban.....	65
7. Hatékonyságvizsgálat és eredményei.....	66
7.1. Fájlonverzió (WDC).....	66
7.2. Hibafelismerési modell (ERM).....	69
7.3. Spreadsheet Lego (Sprego).....	75
7.3.1. Sprego és a hosszútávú tudás kialakítása.....	77
8. Összefoglaló.....	82
9. Jövőbeli elképzelések.....	89
10. Köszönetnyilvánítás.....	89
11. Summary.....	91
12. Irodalomjegyzék.....	95
13. Melléklet.....	106
14. Publikációs jegyzék.....	113

## 1. Bevezetés

A mai kor embereit az információs társadalom kifejezéssel azonosítjuk, mivel az emberek mindennapi életében kulcsszerepet játszik az információk rögzítése, megjelenítése, kezelése és keresése. Ezeknek a feladatoknak az elvégzéséhez azonban a közismereti informatikai ismereteken túl, számítógépes gondolkodásra és algoritmikus készségekre és digitális írástudásra van szükség. Ezeknek a képességeknek az elsajátításának a legkézenfekvőbb helye a közoktatás, ahol tanári útmutatással kialakítható egy hatékony problémamegoldó szemléletmód. Az alap- és középfokú informatikaoktatás fókuszában ma elsődlegesen az alkalmazói ismeretek oktatása áll, alkalmazkodva a munkaerőpiaci igényekhez, hiszen ezen programok ismerete és használata szinte minden munkakörben elvárás. Mindezek ellenére a végfelhasználók által elkészített szöveges és táblázatos dokumentumok többsége hibás (Csernoch & Biró 2015b, Panko 2008, 2010). Ennek hátterében a próbálgatáson, barkácsoláson alapú (Ben-Ari 1999) felhasználói attitűd, valamint a digitális bennszülöttek (Prensky 2001; Jukes & Dosaj 2006, Szabados 2009; Jukes et al. 2010) vélt informatikai tudása áll, akik együtt nőttek fel az okos- és digitális eszközökkel. Ezt a jelenséget erősíti, hogy nincs egyetértés a digitális generáció szervezett informatikaoktatásának szükségességét tekintve. A generáció tagjairól úgy tartják – valamint egyesek maguk is úgy vélik –, hogy nincs szükségük informatikaoktatásra, mivel anyanyelvi szinten értik és beszélik a digitális technológiák, számítógépek, játékkonzolok és az internet nyelvét. Az utóbbi években több tanulmány is bizonyította, hogy attól, hogy valaki e generáció tagja, nem jelenti automatikusan, hogy rendelkezik magas szintű digitális műveltséggel és számítógépes gondolkodással (Prensky 2001; Kirschner & Bruyckere 2017;

U. S. Department of Education 2018). A hanyag felhasználói viselkedést tovább erősíti a szoftvergyártó cégek „felhasználóbarát” felülete. Az alkalmazói szoftverek esetében 2-3 évente újabb verziók jelennek meg a piacon, amelyek újabb és újabb bővítést, kényelmes, problémaspecifikus megoldásokat kínálnak a felhasználók számára. Ezek a kényelmes megoldások azt erősítik a felhasználókban, hogy hagyatkozzanak a szoftverek kínálta, adott problémákra kidolgozott megoldásokra. A modern külsőbe öltöztetett felület gyors megoldásai adnak téves magabiztosságot a felhasználóknak, hogy akár iránymutatás nélkül képesek helyesen használni ezeket a programokat, hiszen minden funkció egy-két kattintásra elérhető. Azonban a gyors és kényelmes megoldások nem jelentenek pontos és precíz munkavégzést, ezért kiemelten fontos a közoktatás szerepe. Az informatikaoktatás fontosságát igazolja az is, hogy a számítógépes gondolkodást a negyedik alapkészségként tartják számon (Kirschner et al. 2006; Wing 2006) az írás, az olvasás és a számolás mellett. Ezek az alapkészségek, a nevükből adódóan, képezik az oktatási programok alapját, melyek elengedhetetlenek a jó iskolai teljesítményhez. A három alapkészség esetében a hozzájuk kapcsolódó tantárgy létjogosultsága megkérdőjelezhetetlen, ezek a tárgyak végig kísérik a tanulókat az egész közoktatás folyamán, így a számítógépes gondolkodás fejlesztésének esetében is hasonló döntésre várnunk.

### **1.1. Célkitűzés**

2012-ben kezdtem meg a tudományos kutatómunkát a TAaAS (Testing Algorithmic and Application Skills = Algoritmikus és Alkalmazási Képességek Tesztelése) (Biró et al. 2015) projekthez csatlakozva, amely 2011-ben a Debreceni Egyetem 4586 hallgatójával végzett informatikai vizsgálatot. A projekt egyik részterületének alapos vizsgálatát és elemzését vállaltam magamra, amely a tanulók informatikai ismereteivel foglalkozik, ezen belül pedig az

egér- és billentyűzet használatával. A feladatlapok javítása, értékelése közben megdöbbenett, hogy milyen alacsony informatikai ismeretekkel érkeznek az egyetemre az első éves hallgatók, függetlenül attól, hogy rendelkeznek-e ECDL vizsgával, vagy informatika érettségivel. Ez a kettősség indította el a tudományos pályafutásomat, valamint a pedagógusi pálya iránti érdeklődésemet. Azóta is töretlen célom, hogy keressem az alacsony tanulói tudás háttérben álló okokat. Az okok keresése mellett a probléma megoldására is nagy hangsúlyt fektettem. A kutatócsoport tagjaival együtt dolgozva különböző informatikai témakörökhöz hatékony, problémamegoldáson és sémákon alapuló oktatási módszereket fejlesztettünk. A módszereket folyamatosan alkalmazzuk és teszteljük a gyakorlatban, hogy bizonyítsuk a feltevéseinket. A dolgozatomban ezek az ok-keresések, a kidolgozott módszerek, az elvégzett felmérések és azok eredményeit mutatom be.

## 2. Tézisek

**[T1] A séma és algoritmusalapú módszertanok (WDC<sup>1</sup>, ERM<sup>2</sup>, Sprego<sup>3</sup>) szignifikánsan hatékonyabbak, mint a hagyományos, felületalapú megközelítések.**

A kutatócsoport tagjaival az informatikaoktatás három témaköréhez – fájlkezelés, szövegszerkesztés és a táblázatkezelés – fejlesztettünk séma-alapú algoritmusépítésre épülő módszereket. Mindhárom témakörben a módszerek

---

<sup>1</sup> WDC – webtable-datatable conversion angol kifejezés rövidítése, a fájlkezelés oktatására használt oktatási módszer, melynek magyar megnevezése: Fájlkonverzió.

<sup>2</sup> ERM – error recognition modell angol kifejezés rövidítése, a szövegkezelés oktatására használt módszer, melynek magyar megnevezése: Hibafelismerési Modell.

<sup>3</sup> Sprego – Spreadsheet Lego angol kifejezés rövidítése, a táblázatkezelés oktatására használt módszer.

tanórai alkalmazása mellett hatékonyságvizsgálatot végeztünk a közoktatás szereplőivel. A módszerek vizsgálata során a módszertanok által kialakított számítógépes gondolkodást, az algoritmikus szemlélet kialakulását és változásait követtük, azonos előismerettel és tudásszinttel rendelkező csoportokban. Azon túl, hogy ezek a módszerek az ismeretek átadása során a tanulók sémákban való gondolkodását fejlesztik, kiemelt előnye, hogy a témaköröket egymáshoz kapcsolja, hagyatkozik a korábbi ismeretekre, ezáltal támogatja a tudástranszfert tantárgyon belül, és igényli a tantárgyközi kapcsolatokat is.

**[T2] Az informatikán belüli tudástranszferalapú oktatás hatékonyabb, mint az egymástól független témakörök tanítása.**

A tantárgyon belüli tudástranszfert kiemelten fontos tényezőként kezeljük. Hipotézisünk alapján órakerettől függetlenül a témakörök összekapcsolása felerősítik egymást. A fájlkezeléshez tartozó ismeretek nem csak különálló témakörként vannak jelen, hanem a tanulók minden gyakorlati feladat során alkalmazzák azokat, így az elmélyülésre, rögzülésre sokkal magasabb óraszám áll rendelkezésre, mint a Kerettantervben meghatározott (OFI 2008, 2012, 2020). A 2012-es Kerettanterv (OFI 2012) drasztikusan lecsökkentette az informatika tanórák számát, így a pedagógusok különböző módszereket alkalmaztak az idő megtakarítására, melyek között az egyes témakörök teljes elhagyása is előfordult. A Kerettantervre épülő hivatalos tankönyvekben (Lakosné Makár et al. 2019a, 2019b, 2019c) minden új eszköz bevezetésekor annak felületi ismertetésével, a programablak részeivel indít, ami felesleges ismeretanyag, hiszen ezt a tanulók az operációs rendszer és a fájlkezelés témakörében elsajátították. Ehhez szorosan kapcsolódik az az ismeretanyag, hogy az operációs rendszer egy egységes felületet biztosít, így az ablakok felépítése azonos, így az Office alkalmazói programjainál sincs lényegi eltérés.

A kutatócsoport által alkalmazott módszerek a számítógépes gondolkodás fejlesztésén túl, a sémaépítésen át minden esetben megalapozzák a következő témaköröket. Ahogy a fájlkezelés során alkalmazott fájlkonverziós módszer a szövegszerkesztést és a táblázatkezelést készíti elő, úgy a táblázatkezelés oktatására alkalmazott Sprego módszer, az általa hatékonyan fejlesztett számítógépes gondolkodással lefekteti a programozás és adatbáziskezelés témakörök alapjait. Az algoritmizálási és problémamegoldási képességek fejlesztésével a tanulók elsajátítanak egy olyan szemléletmódot, melyet iskolán kívüli problémák esetében is hatékonyan tudnak alkalmazni.

**[T3] A 9. osztályos középiskolai tanulók nem vagy alig tudnak támaszkodni az általános iskolából hozott ismereteikre fájlkezelői, szövegkezelői témakörökben.**

A Magyarországon alkalmazott spirális oktatási rendszerből adódóan a tanulók egy témakörrel több évfolyamon is foglalkoznak. Az elméletben jól működő rendszer szerint elég évfolyamonként néhány órát fordítani az előzőekben már tanult témakörökre és azt az ismeretet kibővíteni. A gyakorlatban azonban, a tanulóknak nem elég heti 1 óra beosztással 5 alkalom egy témakörre. Nincs idő gyakorlásra, az ismeretek elmélyítésére, így amikor egy év múlva ismét sor kerül a témakörre a tanulók már nem emlékeznek a korábban tanultakra. Az újabb rendelkezésre álló óraszám nem a korábbi ismeretek bővítésére szolgál, hanem azok ismétlésére. A középiskolában nem csak a tanítandó témakörök sora kezdődik előlről, hanem az oktatásuk is a nulláról, annak ellenére, hogy valamennyi témával a tanulók már általános iskolában 3 éven keresztül foglalkoztak. A rosszul felépített tantervek és a rosszul megválasztott oktatási módszerek állnak a tanulók felszínes tudásának hátterében, a tanulók

nem tudatosan használják az informatika eszközeit, nem tudják saját szavaikkal megfogalmazni, megnevezni az elvégzett műveleteket, valamint nem ismerik az informatikához, számítógéphasználathoz kötött terminológiát.

#### **[T4] A tanulók az informatikai ismereteiket az önértékelés során túlértékelik**

A tanulók az informatikai témaköröket egy-egy szoftverkörnyezettel azonosítják. Az informatikai ismeretüket elsősorban ezen szoftverek ismeretével, azok felületén történő navigációval teszik egyenlővé. Célunk az volt, hogy megtudjuk, hogy a tanulók mennyire reálisan tudják értékelni saját informatikatudásukat. A tézis bizonyítását országos szintű adatgyűjtés előzte meg – mini kompetencia teszt –, ahol a tanulók 0-5 skálán értékelhették tudásukat. A kapott önértékelési eredményeket azonos tudással rendelkező tanulók gyakorlati tesztfeladatokban elért eredményeivel hasonlítottuk össze.

### **3. Informatikaoktatás helyzete Magyarországon**

#### **3.1. Tartalmi szabályozók**

Magyarországon a közoktatást szabályozó legfontosabb dokumentumok a Nemzeti Alaptanterv (OFI 1995, 2012a, 2020) és a ráépülő Kerettantervek (OFI 2008, 2012b). A NAT a tanulási tartalom elméleti és fogalmi alapjait biztosítja, az iskolák autonómiáját szem előtt tartva határozza meg az általános célokat, a fő kompetencia területeket (kulcskompetenciák) és a fejlesztési feladatokat. Erre alapozva az Oktatási Hivatal készíti el a Kerettanterveket, amelyek meghatározzák a kötelező tantárgyakat, óraszámokat, az iskolai tantervek tartalmát, és a kimeneti követelményeket – iskolatípustól, képzési programtól

függően. A NAT és a Kerettantervek alapján az iskolák elkészítik a helyi tantervüket, amelyek a tudásanyag logikai felépítését írják le óráról órára.

A számítástechnika önálló területként 1995-ben jelent meg. Ebben az időben azonban nem volt informatikus tanárképzés, melynek következtében át-képzéseket szerveztek, ahol posztgraduális képzéseken, tanfolyamokon a korábbi matematika, fizika és további természettudományos tárgyat oktató pedagógusok váltak informatikatanárokká előzetes informatikai háttértudás nélkül.

Az első számítástechnikát is tartalmazó NAT kiemelten fontosnak tartotta az elemző gondolkodás képességének fejlesztését és a tudatos eszközhasználat kialakítását. Ez a cél azonban időközben elveszett, pontosabban csak az informatikaoktatás szűk részére korlátozódott: a programozás oktatására. Ezzel szemben az alkalmazói ismeretek oktatása során a felület alapú, low-mathability<sup>4</sup> megközelítések nyertek teret, ahol elsődlegesen a szoftver interfészen történő navigációja került fókuszba (Baranyi & Gilányi 2013; Chmielewska et al. 2016). Ez a megközelítés nem csak az alacsony hatékonyság és szoftverkörnyezet függő tudás miatt hátrányos, hanem mert nem alakítja ki a tanulóknál az igényt a megoldás ellenőrzésére sem. A szövegkezelés kapcsán elsősorban a WYSIWYG – What You See Is What You Get = amit látsz, azt kapod – gondolkodásmód van jelen, ahol a felhasználónak az számít, hogy nyomtatásban hogyan fog kinézni a dokumentum, függetlenül attól, hogy jólformázott<sup>5</sup>-e. Az adat- és táblázatkezelés során a felhasználók a TAEW-típusú (trial-and-error wizard-based) – próba-hiba – módszert alkalmazzák, azaz a próbálgatással igyekeznek bármiféle eredményt kapni, nem törődve az eredmény helyes-

---

<sup>4</sup> A metakognitív számítógépes problémamegoldási megközelítések osztályozásra kerültek, a mesterséges és a természetes kognitív képességek matematika szempontjából való vizsgálata által, mely szerint a koncepció-alapú módszerek – amilyen a Pólya-féle módszer is – „matematikussága” a legmagasabb (5), míg a TAEW próbálgatásra épülőké a legalacsonyabb (1) (Baranyi & Gilányi 2013; Borus & Gilányi 2013).

<sup>5</sup>a változtatásokkal szemben invariáns

ségével. Ezek a módszerek nem megbízhatóak és nagy mennyiségű hibát generálnak, amely komoly anyagi károkhoz vezethet nem csak a vállalati szférában (Teo & Tan 1999; Tort et al. 2008; Panko 2008; Powell et al. 2008, 2009a, 2009b; Abraham & Erwig 2009; Kadjevich 2009, 2013; Panko & Aurigemma 2010; Tort 2010; W2 2012; Jorgensen 2013; Kwak 2013; Garrett 2015; Eusprig 2020).

A helytelen oktatási módszerek mellett tovább nehezítette az informatikaoktatás hazai helyzetét a 2012-ben kiadott NAT-ra épülő, 2013-ban bevezetett kerettanterv (OFI 2012a, 2012b), amelyben az informatika órák kötelező óraszámát drasztikusan csökkentették a tananyagtartalom változtatása nélkül. A 2013-ban bevezetett kerettanterv nagyon sok kritikát kapott a pedagógusok részéről, mivel a tananyag mennyisége és a rendelkezésre álló időkeretet nem volt összhangban (Sebestyén 2014). Mások a tananyagtartalom megfogalmazását találták túl általánosnak, amelyet minden pedagógus eltérően értelmezett, attól függően mennyire mély ismeretekkel rendelkezett a témakörben. Nemzetközi tapasztalatokra épülve, a sikeresebb informatikaoktatás reményében az új kerettantervben (OFI 2020) ismét növelték a kötelező óraszámokat a korábbi évekhez képest. Továbbá a 2019 végén kiadott NAT szerzői előtérbe helyezték a problémamegoldás fontosságát, így az már nem csak a programozásban, hanem több témakörben is hangsúlyosabban megjelenik, azonban azok mindegyike szorosan kapcsolódik a programozásoktatáshoz.

Az egyik legfontosabb kérdés, hogy ezek a változások – órakeret emelése, problémamegoldás erősítése, témakörök átszervezése – a gyakorlatban hogyan fognak megvalósulni? Az oktatási rendszer nem biztosított sem módszert, sem továbbképzési lehetőségeket a probléma-alapú oktatási stratégiák elsajátítására, így a pedagógusok a megszokott módon, felületi megközelítéssel folytatják az oktatást. A magasabb óraszám pedig nem fogja beváltani a hozzá fűzött

reményeket, mivel a felületen való navigáció, kattintgatás egyrészt nem fejleszti a tanulók számítógépes gondolkodását, sem pedig a rendszerezett és hosszantartó tudás kialakulását.

### 3.1.1. Alkalmazói ismeretek fontossága és a NAT

A NAT-ban (OFI 2020) központi szerepet kapott a digitális kompetencia, a digitális írástudás, amely kifejezések többször is szerepelnek és nem csak a digitális-kultúra szabályozójában.

„Fontos cél a digitális kompetencia fejlesztése is, az IKT-eszközök tudatos és kreatív alkalmazása.” (NAT - Magyar nyelv és irodalom 9–12. évfolyam)

Ennek létjogosultságát mi sem tükrözi jobban, mint a helytelen szerkesztésű NAT 2020 (1. ábra, 2. ábra). Az 1. ábra hibái – fejlécben üres bekezdés, cím középre igazítás helyett számtalan szóközzel került nagyjából középre, amit újabb üres bekezdés követ – igazán feltűnőek, hiszen ezek a dokumentum első sorai, a megnyitás után egyből ezekkel szembesülünk.



¶

.....9-12. ÉVFOLYAM.¶

¶

Az anyanyelv már nevében is a legszorosabb összetartozást fejezi ki az azonos nyelvet beszélő emberek között. Az anyanyelven megszülető irodalom alkotói és hallgatói olyan olvasói

1. ábra A NAT első oldala a Magyar nyelv és irodalom 9-12. évfolyam szabályozóban, amelyben több tördelési hiba is szerepel. (OFI 2020)

A 2. ábra újabb és újabb tördelési hibákat mutat be. A táblázat első oszlopában az A) és B) pontok kézi számozással készültek, míg a C) és D) pontok esetében automatikus a számozás. A bekezdések behúzása az 1. ábra címéhez hasonlóan szóközzel történt. Mindezek mellett a sorok végén is megjelenik

az Enter, a bekezdés vége jel, annak ellenére, hogy a mondat a következő bekezdésben folytatódik (pirossal keretezett részek).

<b>I. → Bevezetés az irodalomba – művészet, irodalom</b>	
.....A) Az irodalom és hatása	
.....Karinthy Frigyes: A cirkusz	Örkény István: Ballada a költészet hatalmáról II. János Pál pápa levele a művészeknek (részletek)
.....B) Szerzők, művek párbeszéde	
.....Aiszóposz: A tücsök és a hangya	
.....Hajnóczy Péter: A hangya és a tücsök	
.....Romhányi József: Tücsökdal	
C) Népszerű irodalom. Az irodalom határterületei	
.....Arthur Conan Doyle: Sherlock Holmes történetek (részletek)	Irodalom és film Agatha Christie: Tíz kicsi néger
D) Műnemi-műfaji rendszer	

2. ábra Magyar nyelv és irodalom 9-12. évfolyam szabályozójából további tördelési hibák. (OFI 2020)

Ezek a példák jól szemléltetik, hogy mennyire indokolt az alkalmazói ismeretek oktatására is hangsúlyt fektetni, mennyire szükséges szakítani a Magyarországon elterjedt és alkalmazott felületalapú megközelítésekkel és helyette a problémamegoldáson alapuló módszereket előtérbe helyezni, nem csak a programozásoktatás során.

### 3.2. PISA és a digitális szövegértés

A magyar tanulók évről évre részt vesznek a PISA felmérésben, amely elsősorban a tanulók olvasáskészségét és matematikai és természettudományos ismereteit vizsgálja. A 2009-es PISA Student Online (OECD 2011) a tanulók digitális kompetenciáinak mérésére irányult. Ebben a vizsgálatban a résztvevő 19 ország közül a magyar tanulók eredményei a 15. helyre voltak elegendőek. Az általuk elért eredmény szignifikánsan alacsonyabb az OECD átlagtól (OECD 2011). A magyar tanulók digitális szövegértése a 2. szintre tehető (2. szint, 3. szint, 4. szint, 5. szint és fölötté), amit az OECD belépő, minimumszintnek sorol be és a következőképpen definiálja:

“képesé teszi őket az iskolai, közösségi és személyes életben való önálló részvételre...

...nem tudnak információt keresni az interneten csak magukra hagyatkozva, azaz explicit útmutatás nélkül, és kihívást jelent számukra a források hitelességének és relevanciájának kritikus megítélése.” (Oktatási Hivatal 2009)

Az elemzők összevetették a 5. és 95. percentiliseket, melynek során a legjobban és a leggyengébben teljesítő tanulók 5%-ának eredményeit hasonlították össze. A résztvevő 19 ország közül, Magyarország legjobban és a leggyengébben teljesítő tanulóinak eredményei között 337 képességpont volt a különbség, amely a legmagasabb volt mindközül. A felmérés ezzel rámutatott arra is, hogy mennyire egyenetlen, változó színvonalú Magyarországon az informatikaoktatás. A Nemzeti Alaptanterv fő célkitűzése, hogy meghatározzon egy közös műveltségi alapot, valamint lehetővé tegye az iskolák közötti átjárhatóságot. A PISA eredmények ennek ellenkezőjét mutatták, cáfolva a NAT céljának megvalósulását. A felmérés idején, a mintába kerülő tanulók a 2008-as NAT szerint tanultak, így az ehhez tartozó magas óraszám a mérvadó.

További fontos információkat szolgáltatott az iskolai számítógéphasználat és a digitális írástudás kapcsolatának elemzése. A kitöltők válaszai alapján egyedül Magyarországon mutatott ki negatív kapcsolatot a PISA elemzés ezen területek között.

A későbbi, 2012-es mérésben továbbra is szerepelt a digitális szövegértés, azonban már kevesebb feladattal. Ezt követően, a 2015-ös és 2018-as mérésekre megszűnt ez a terület, helyette a korábbi nyomtatott szövegértési feladatokat jelenítették meg digitális médium segítségével. A 2012-es PISA felmérésben a magyar tanulók a 2009-es eredményekhez képest is gyengébben teljesítettek. Emelkedett az alul teljesítő tanulóknak a száma, akik a 2. minimál

szintet sem érték el. A magasabb jártassági szinteket pedig alacsonyabb arányban érték el a magyar tanulók 2012-ben (OECD 2015) (1. táblázat).

A 2009-es és a 2012-es PISA eredmények azt mutatják, hogy a magyar tanulók közel harmada „digitális analfabéta”, annak ellenére, hogy ezt a generációt digitális bennszülöttnek nevezi a szakirodalom (Prensky 2001). Erről a korosztályról úgy tartják, hogy beleszórt a digitális világba, már gyermekként „megtanulták” a számítógépek, okostelefonok és tabletek használatát. Ezt az állítást azonban a PISA mellett más kutatási eredmények is cáfolják (Kirschner & Bruyckere 2017; Prensky 2001). A Debreceni Egyetemen végzett felmérés bebizonyította, hogy az okostelefonok használata nem segíti a számítógépes gondolkodást, valamint az algoritmikus készségek fejlesztését (Biró et al. 2018; Csernoch & Biró 2019). A tanulóknak, annak ellenére, hogy a digitális bennszülött generáció tagjai, szükségük van informatikaoktatásra (Kirschner et al. 2006), valamint szakértő tanári útmutatásra (Hattie 2003, 2012).

	2009	2012	változás
5. szint és afölött	4,84%	4,05%	-0,79%
4. szint	16,31%	14,06%	-2,25%
3. szint	27,05%	24,83%	-2,22%
2. szint	24,97%	24,58%	-0,39%
2. szint alatt	26,82%	32,48%	+5,66%

1. táblázat A PISA 2009 és 2012 évi digitális szövegértés mérésén a magyar tanulók jártassági szintjeinek összehasonlítása (OECD 2011, 2015)

### 3.3. Kihívások, nehézségek a pedagógusok számára

Az elmúlt 10 évben körvonalazódott az egyre fokozódó pedagógushiány. Az idősödő tanárállomány pótlását az állam nem képes megalapozni, a tanárképzés hossza (5 év) és a diplomával járó pedagógusbér (töredéke más diplomás béreknek) nem csábítja a fiatalokat a pályára lépésre. A tanárképzésben résztvevők csatlakozhatnak a Klebelsberg Képzési Ösztöndíj Programhoz (Klebelsberg Központ 2020), melyben a hallgatók az ösztöndíj (differenciáltan

125-250-375 ezer Ft/félév) ellenében vállalják, hogy főállású, pedagógus foglalkoztatásra irányuló jogviszonyt létesítenek és tartanak fent a diploma megszerzését követően. Az ösztöndíj igen kecsegtető lehet a fiatalok számára, azonban nagy hátránya, hogy a képzés elvégzése után nem választható meg az intézmény, ahol szeretnének elhelyezkedni.

Az Európai Bizottság is felfigyelt a Magyarországot érintő tanárhiányra (Európai Bizottság 2020). A 2020 februárjában megjelenő jelentésében első-sorban az alacsony pedagógusbéreket jelöli meg okként. A pályakezdekők bére az egyik legalacsonyabb az EU-ban (Eurydice 2018). A pedagógusképzésben nagy a lemorzsolódás, a végzettek kevesebb, mint fele helyezkedik el a pályán.

A pedagógusok bérének rendezése érdekében 2013-ban bevezetésre került a pedagógus életpálya modell, amelynek követelményrendszere évről évre változott, azonban szakmai és anyagi megbecsülést nem hozott magával. A 2019-ben megszavazott, 2020. július elsején életbe lépő intézkedés értelmében, a szakoktatók bére differenciáltan 30%-kal emelkedett. Az intézkedés elsődleges célja, hogy a szakembereket a tanári pályán tartsák, valamint a piaci szférából is vonzzanak be új oktatókat (Csókás, 2019). Az intézkedés a korábbinál is élesebb bérfeszültséget és ellentéteket okozott a különböző iskolatípusok dolgozói között. További probléma, hogy a béremelés nem a pályán lévő pedagógusok megbecsülését segíti, hanem olyan személyek bevonását (akár másodállásként) szorgalmazza, akik a piaci szférában egyébként is jól kerestek. A beharangozott béremelés nem garantálta a fizetésemelés odaitélését, az országos visszajelzések alapján átlagosan 20%-ra tehető az emelés, azonban vannak kirívó példák – 23 évnyi munkaviszony után 10,6%; a pótlékokat (osztályfőnöki, munkaközösségvezetői) pedig a béremelés részeként tüntették fel (Eduline 2020, Pénzcentrum 2020).

Az informatika tanároknak az elmúlt évben további nehézséggel is szembe kellett nézniük. A 2019-ben megjelenő COVID-19 vírus következtében 2020.

március 16-tól új, tantermen kívüli digitális munkarendben kezdődött meg az oktatás Magyarországon. A váltásra nem volt felkészülve az oktatási rendszer, sem pedig az előregedett pedagógusok közössége. Az OECD 2018-as felmérésében a magyar pedagógusok átlag életkora 47,6. A megkérdezett 3245 magyar pedagógus csaknem fele 50 év feletti, míg a 30 év alattiak száma az 5%-ot sem érte el (OECD 2018). A digitális munkarend, így extra felelősséget és munkát jelentett az informatikatanároknak, ahol az idősebb generáció tagjainak a tanítása is az ő feladatukká vált.

## **4. Előzetes vizsgálatok és eredmények**

### **4.1. Algoritmikus és alkalmazási készségek tesztelése – TAaAS<sup>6</sup>**

A TAaAS (Bíró et al. 2015) projekt a Debreceni Egyetem első éves hallgatóinak informatikai ismereteinek vizsgálatát tűzte ki célul. A felmérést Csernoch Mária és Bujdosó Gyöngyi a Debreceni Egyetem Informatika Karának oktatói végezték 2011 augusztusában. A projekt a DE rektorának, Dr. Fábíán Istvánnak és dr. Végső János a DE Tudományegyetemi Karok HSZK vezetőjének támogatását élvezte. Ennek a támogatottságnak köszönhető, hogy a felmérésben a DE 15 karának 4586 hallgatója vett részt. A felmérés az első éves hallgatók informatikai ismereteit, számítógéphasználati szokásait térképezte fel. A felmérés több részterületre osztható, ezek rendre: internetes tartalmak, elektronikus tananyaggal kapcsolatos olvasási szokások, regényolvasási szokások, elektronikus helyesírás és informatikai alapismeretek.

---

<sup>6</sup> A felmérés angol megnevezésének rövidített változata: Testing Algorithmic and Application Skills

A projektben a korábban felsorolt részterületek közül a hallgatók informatikai ismereteit feltérképező részterület ellenőrzésében, pontozásában, digitalizálásában és elemzésében vettem részt. A részterület is több informatikai ismeretet érint, ezek közül a billentyűzet– és egér használattal kapcsolatos feladatokra fókuszáltam (3. ábra, 4. ábra).

A szakmaiságot szem előtt tartva, kérjük, határozza meg a kért funkciókat és fogalmakat! A felhasználói környezetet az egyes feladatoknál külön jelöljük.

1. Határozza meg a következő billentyűk funkcióját szövegszerkesztői környezetben (pl. Jegyzettömbben, MS Wordben, OpenOffice Writerben, WordPadben)!
  - a) Insert .....
  - b) Home .....
  - c) End .....
  - d) Page Up .....
2. Hasonlítsa össze a Delete és a Backspace billentyűk használatát szövegszerkesztői környezetben (pl. Jegyzettömbben, MS Wordben, OpenOffice Writerben, WordPadben)!
  - a) Eltérés .....
  - b) Hasonlóság .....

3. ábra TAaAS felmérés Informatikai ismeretek részterületének 1. és 2. feladata, amelyek a billentyűzet használati ismeretekhez kapcsolódnak.

10. Magyarozza el, milyen hatása van az alábbi egérekattintásoknak az előző feladatból már megismert, a kattintás mellett megadott elemeken! A válaszhoz vegye alapul a Windows operációs rendszerek alapértelmezett beállításait!

Egér gombja	Kattintás száma	Kiválasztott elem	Hatás
bal	egy	 olimpia	
bal	dupla	 olimpia	
bal	kettő	 olimpia	
bal	dupla	 manye_2009.ppt	
jobb	egy	 manye_2009.ppt	
bal	dupla	 tcmd750a.exe	
bal	egy	 kek.bbb	

4. ábra A TAaAS felmérés Informatikai ismeretek részterületének 10. feladata, amely a hallgatók egérhasználatához kapcsolódó terminológiai tudására összpontosít.

Mind a billentyűzet, mind az egér azon perifériák közé tartozik, amelyek részei a mindennapi számítógéphaszúlatnak. Ezeket a beviteli eszközöket már az első perctől kezdve használják, ismerik a felhasználók. A 3. ábra és 4. ábra feladatai a tudatos eszközhasználat mellett azt vizsgálja, hogy a hallgatók mennyire képesek helyesen használni az eszközhasználatához kötődő szakmai

nyelvet. A feladatok által bepillantást nyerhetünk, hogy a hallgatók a beszélt nyelvi kommunikációban mennyire képesek információt közölni és befogadni.

A billentyűzet és az egér használata nyilvánvalónak tűnik, és nem csak a digitális bennszülöttek (Jukes & Dosaj 2006, Jukes & McCain & Crockett 2010) esetében, így nem meglepő, hogy az informatikai tankönyvek közül csupán néhány foglalkozott – a hivatalos tankönyvcsalád bevezetése előtt – ezekkel az eszközökkel, azonban ezek sem egységesek. Egymástól nagyban eltérő billentyű leírásokat és csoportosításokat, valamint következtelen terminológiai használatot találtunk a tankönyvek elemzése során.



5. ábra Billentyűzet csoportosítása az *Informatika 5. osztály* című könyvben. (Bánné Mészáros et al. 2008)

Voltak olyan tankönyvek, segédanyagok, amelyek a billentyűket a feladatok szerint csoportosítják, azonban ebben sincs egyetértés, sem az egyes csoportokba tartozó billentyűkről, sem pedig a csoportok elnevezéséről (5. ábra, 6. ábra, 7. ábra).

A csoportosítás azonban nem jelent meg minden könyvben, erre példa *Számítógéphasználat mindenkinek* című könyv, ahol a billentyűk ismertetése rendezetlenül, véletlenszerű sorrendben követik egymást (Bártfai 2011).

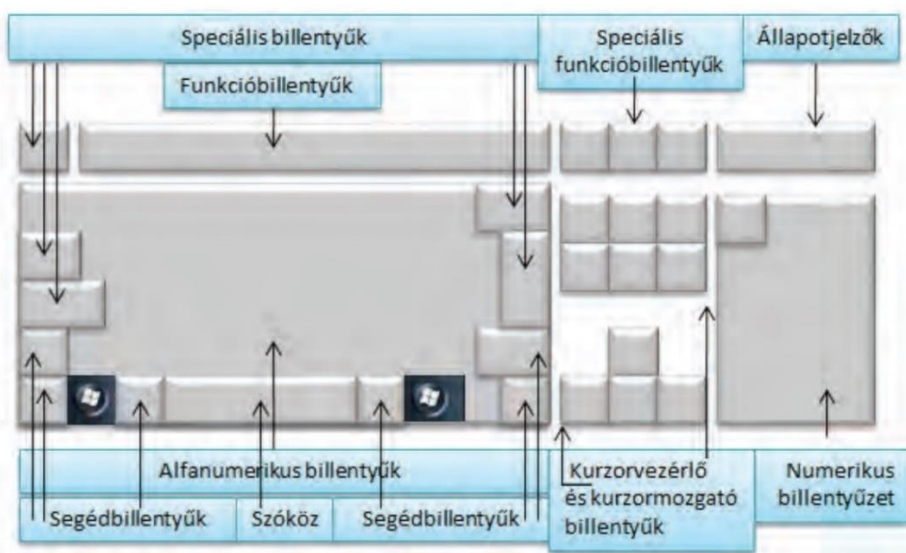
2013 óta az állam megtagadta több szakmai kiadó tankönyvének engedélyét, ezzel alapozva meg a hivatalos egyentankönyv létrejöttét. A tankönyvki-

adók helyzetét az állam folyamatosan megnehezítette, 5 évente volt csak megengedett a tankönyvek tartalmának frissítése és átdolgozása, ez azonban az informatika területén nagyon hosszú idő. A rendelkezés miatt volt szinte minden tankönyv tartalma és illusztrációja elavult (Dercsényi 2016).



6. ábra Billentyűzet csoportosítása a gyártó saját oldalán (Microsoft 2020).

Az egyik hivatalos tankönyv (Lakosné et al. 2019a) billentyűzete és csoportosítása látható a 7. ábra képén.



7. ábra Billentyűzet csoportosítás az Informatika 6. című könyvében, amely 2019 óta a hivatalos informatika tankönyvcsalád tagja. (Lakosné et al. 2019a)

A másik két példához képest (5. ábra, 6. ábra) a csoportok jelölése kaotikus, ezt tovább tetézi, hogy az ábrához tartozó billentyűk leírása csoportokban történik, azonban az nem teljesen egyezik meg a képen láthatóval.

A forgalomban kapható informatika tankönyvek és segédanyagok vizsgálata során azt tapasztaltuk (egyentankönyvek bevezetése előtti tankönyveket is beleértve), hogy nincs egységes terminológia és állásfoglalás a billentyűk csoportosításával kapcsolatban, az egérhasználathoz kötött terminológia szinte teljes egészében hiányzik a tankönyvekből. Ez részben lehet magyarázata a TAaAS felmérésben résztvevő hallgatók kirívóan alacsony eredményeire, de nem tehető egyedül felelőssé. Az alacsony eredményeken túl (2. táblázat) kiváltképp meglepő az egyes karok teljesítménye.

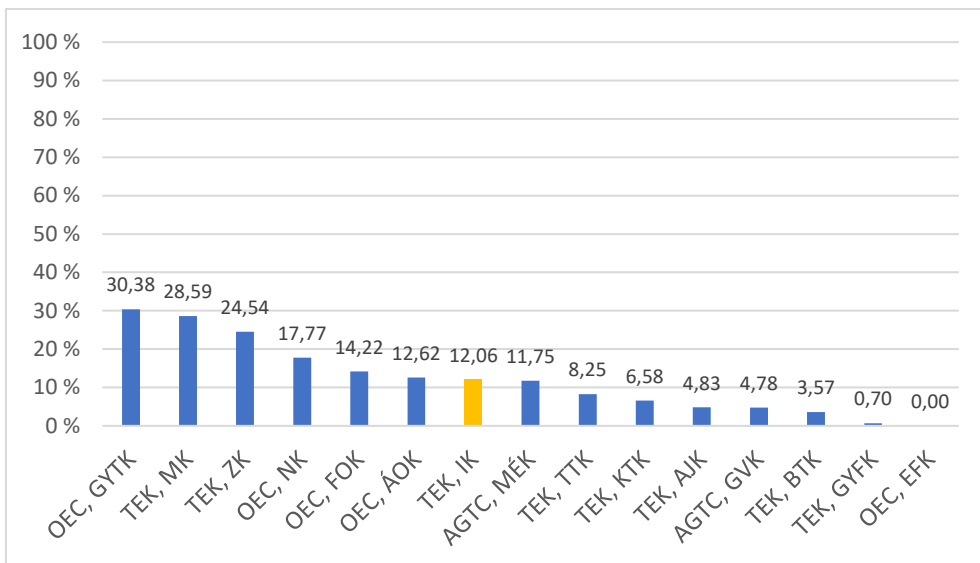
		össz.	IK	legjobb eredmény	legjobb eredményt elérő kar
billentyűzet	insert	0,74%	2,25%	2,25%	TEK, IK
	home	1,90%	3,75%	5,66%	OEC, NK
	End	2,12%	4,00%	5,66%	OEC, NK
	Page UP	0,22%	0,25%	5,56%	TEK, ZK
	Delete/Backspace különbség	1,85%	1,50%	5,02%	TEK, MK
	Delete/Backspace hasonlóság	23,72%	26,25%	74,19%	OEC, GYTK
egér	Egy kattintás bal gombbal mappára	21,26%	24,00%	66,51%	TEK, MK
	Dupla kattintás bal gombbal mappára	22,66%	23,50%	70,97%	OEC, GYTK
	Kettő kattintás bal gombbal mappára	12,32%	15,00%	54,84%	OEC, GYTK
	Egy kattintás jobb gombbal	3,99%	10,75%	11,11%	TEK, ZK
	Dupla kattintás programfájlra	7,94%	14,50%	38,71%	OEC, GYTK
	Egy bal kattintás nem-társított adatfájlra	14,04%	19,00%	45,45%	TEK, MK

2. táblázat A TAaAS felmérés összesített eredménye feladatokra bontva. Az össz. oszlopban a 15 kar együttes eredménye szerepel, mellette az IK oszlopban az Informatika Karra felvételt nyert hallgatók eredménye. Szerepel a legjobb eredmény a karok közül, valamint a Kar neve, amihez az az eredmény tartozik. Az Intézmények és a Karok rövidítésének feloldása a következőképpen alakul, TEK – Tudományegyetemi Karok, OEC – Orvos és Egészségtudományi Centrum, a továbbiakban IK – Informatikai Kar; NK – Népegészségügyi Kar; ZK – Zeneművészeti Kar; MK – Műszaki Kar; GYTK – Gyógyszertudományi Kar.

Az Informatikai Kar hallgatóitól vártuk a legmagasabb eredményeket, hiszen ők érdeklődnek leginkább az informatika iránt, valamint ők rendelkeznek

elsősorban közép- és emelt szintű informatika érettségivel és/vagy ECDL bizonyítvánnyal. A részfeladatok alapján egy kivétellel minden esetben megelőzte az IK hallgatóit valamelyik másik kar, kiemelten jól szerepeltek az Orvos és Egészségtudományi Centrum Népegészségügyi Kara (OEC, NK) és Gyógyszertudományi Kara (OEC, GYTK), emellett a Tudományegyetemi Karok közül a Műszaki (TEK, MK) és a Zeneművészeti Kar (TEK, ZK) (2. táblázat).

Az eredmények karonkénti összesítése alapján a korábban kiemelt karok (GYTK, NK, MK, ZK) érték el a legjobb eredményeket (8. ábra). Az Informatikai Kar a középmezőnyben teljesített. A Karok hallgatói eredménye (2. táblázat) és illetően sorrendje (8. ábra) megkérdőjelezi az informatikaoktatás hatékony működését és az informatika érettségi és az ECDL bizonyítvány megszerzéséhez szükséges informatikai ismereteket, valamint azok hosszútávú memóriába való beépülését. A TAaAS felmérés eredményei indítottak el az okok és megoldások keresésének útján. Azonban az oktatás hibára nem elég csak rámutatni, hanem azok megszüntetésének, kiküszöbölésének érdekében konstruktív javaslatot kell tenni. A kutatócsoporthoz csatlakozva, így olyan oktatási módszerek, megközelítések alkalmazását kezdtük kidolgozni, amelyek elősegítik a hatékony informatikaoktatást. Az akkor már szárnyát bontogató Sprego (táblázatkezelői) módszerre fókuszáltunk, mely módszer hatékonyságát a TAaAS-t követő években elsősorban az Informatikai Kar hallgatóin teszteltük, majd később Csapó Gábor és általam a közoktatásba is sikeresen beépült, ahol a módszer hatékonysága szintén bizonyításra került (Csapó et al. 2020, 2021). A kutatócsoport azóta is a tudástranszfer elemek kiaknázásában, a sémaépítésben látja a hatékony informatikaoktatás kulcsát, így ezekre épülő módszereket fejlesztett és alkalmazott, hogy változtasson a 2011-es TAaAS felmérés gyenge hallgatói eredményein.



8. ábra A TAAAS felmérés billentyűzet- és egerhasználatát vizsgáló feladatainak karonkénti eredménye csökkenő rendezésben.

## 5. Módszerek

Az informatikaoktatás szakmai köreiben – a hatékonyság jegyében – a tanulók algoritmikus gondolkodásának fejlesztése került a középpontba. Nemzetközi szinten már több kutató és tanulmány rámutatott a problémamegoldás és a sémaépítés fejlesztésének fontosságára, amelyet az algoritmizálás ismételt fókuszba helyezésével tartják megvalósíthatónak (Hermans 2019; Hubwieser 2004; Kahneman 2011; Sestoft 2011; Sweller et al. 2011; Swidan & Hermans 2019). Az algoritmus, a problémamegoldás az elmúlt 3-6 év hívószava a hazai informatikatanárok éves konferenciájának is, ahol egyre több előadás és publikáció születik a témakörben (Webdidaktika Alapítvány 2017, 2018, 2019). Több kutató felismerte, hogy ez a fajta gondolkodásmód kialakítása kortól független, így már egészen fiatal korban el kell és lehet kezdeni a fejlesztését (Bernát 2015; Bernát & Zsakó 2019; Szabó 2017). Nagy hangsúly kerül már az

óvodai oktatásra, ahol különböző robotokkal játékos formában fejlesztik a gyerekek gondolkodásmódját. Azonban ezek a kezdeményezések gyakran gyerekcipőben járnak, a módszerek hatékonyságvizsgálatára kevés esetben került sor, inkább egyfajta „legjobb gyakorlat”-ként kerülnek bemutatásra. A konferenciák tartalma, a tanulmányok eredményei elsősorban a résztvevők számára válik ismertté, akik valamennyien a hatékony informatikaoktatást tűzték ki céljukul, így őket meggyőzni a szemléletváltás fontosságáról már nem kell. A problémamegoldáson alapuló módszerek, megközelítések ilyen módon nem érnek célba, nem teszik lehetővé új pedagógusok bevonását, képzését.

### **5.1. Problémamegoldás**

Problémának tekintünk minden olyan szituációt, helyzetet, amikor valamilyen akadály áll a cél elérésének útjában (Lénárd 1984), amely egybehangzik Pólya György (1954) meghatározásával, mely szerint a probléma megoldása egyfajta útkeresés. Több problémamegoldási modell létezik (Anderson 1993, De Groot 1956, Dewey 1910, Hutchinson 1949, Johnson 1972, Lénárd 1978, Newell et al. 1962, Owen & Sweller 1989, Pólya 1954; Skemp 1971, Sternberg 1980, Vinacke 1952, Wallas 1926, Young 1940), azonban a legtöbb alkalommal a Pólya (1954), valamint Wallas (1926)-féle modelleket és annak variánsait alkalmazzák. A koncepcióalapú problémamegoldás nem más, mint a megfelelő lépések sorrendje, amelyet Pólya egyértelműen meghatározott a *How to solve it* (1954) című könyvében. Pólya György 4-lépéses modellje a következőképpen néz ki:

1. lépés: probléma megértése, cél meghatározása
2. lépés: a terv kidolgozása
3. lépés: a terv végrehajtása
4. lépés: ellenőrzés, diszkusszió

Az első lépések egymásra épülnek, a probléma megértése nélkül a cél sem határozható meg, hogy mit szeretnénk elérni. Ehhez szorosan kapcsolódik az ismert adatok elemzése és hasonló probléma keresése. Készítsünk tervet a megoldáshoz, ábrázoljuk és határozzuk meg a szükséges eszközöket és módszereket. Hajtsuk végre, ellenőrizzük, hogy valóban az első lépésben meghatározott célt értük el. Ellenőrizzük és járjuk körbe a kapott megoldást.

A sikeres problémamegoldásnak feltétele a kritikai és a kreatív gondolkodás (Wachsmuth 1981), így a kutatócsoport az oktatási folyamat során az eltérő gondolkodási stratégiákat fejleszti és hagyja érvényesülni. A Pólya-féle kutatás alapú tanulás technikája jól alkalmazható a tanulók kreatív és kritikai gondolkodásának ösztönzésére. A tanórákon a tanulók természetes nyelven fogalmazzák meg ötleteiket, nincsenek rossz ötletek, minden javaslatot értékel, elemez a csoport, következtetéseket von le, vagyis nagy szerepet kap a hibákkal szembeni konstruktív hozzáállás.

Pólya problémamegoldó megközelítése felismerhető a 2013-as IEEE&ACM jelentésben (IEEE&ACM Report 2013), amely három egymásra épülő informatikai tudásszintet határoz meg (1. ismertség – probléma megértése, 2. felhasználás – kivitelezés, 3. értékelés). A jelentésben szereplő tudásszintek megfeleltethetők a Pólya-féle megoldási stratégia 1., 3. és 4. lépésének. A tudásszinteket tekintve a hazai informatikaoktatás a második szintre fókuszál, a végrehajtásra. Mellőzi a probléma megértését és a terv készítését, ennek eredményeként a harmadik szint elérésére, a konklúzió levonására, az általánosításra nincs lehetőség. A hazai közoktatásban általánosan elterjedt és alkalmazott hagyományos felületalapú módszerekre jellemző a minta és/vagy szakácskönyv alapján történő feladatmegoldás az alkalmazói ismeretek oktatása során. A szakácskönyv jellegű feladatokban a megadott lépések végrehajtására (2. szint), tehát az alkalmazói szoftver felületének ismeretére van szükség, míg

a minta alapú feladatok esetében nem számít a választott eszköz, módszer, csupán a dokumentum kinézete, amely nem fejleszti a tanulók számítógépes gondolkodását (Kirschner et al. 2016).

## 5.2. Sémaépítés

A programozásoktatásban a kezdetektől megjelenik a séma-, és a sémaépítés fontossága (Szlávi et al. 2019). Ennek a gondolkodásmódnak az elsajátítása elengedhetetlen a hosszútávú tudás kialakításához és a hatékony problémamegoldás eléréséhez.

Nemzetközi szinten az utóbbi évtizedekben az oktatás hatékonyabbá tétele és a különböző oktatási módszerek iránt megnőtt az érdeklődés. Az oktatás szereplői nem csak a hatékony(abb) módszerekben érdekeltek, hanem hogy a tanulók, hallgatók kritikus gondolkodóvá váljanak. Az ember a mindennapi élet kihívásainak nem tud megfelelni kritikus gondolkodás nélkül, az a sok inger és információ, az összetett problémák, a gyors technikai és társadalmi változások a kritikus gondolkodás és a digitális írástudás meglétét megkövetelik. A kutatók a kritikus gondolkodást állítják a középpontba az oktatás minden szintjén (Adaboh 2016, Merriënboer & Sweller 2005). Az 1980-as évektől vált ismertté a Cognitive Load Theory – „kognitív terhelés elmélet” –, amely két féle memóriát különböztet meg, hosszútávú- és munkamemóriát. A tanulási, oktatási folyamat során a cél, hogy az ismeretek a hosszútávú memóriába kerüljenek, ehhez pedig a rövidtávú munkamemórián keresztül lehet eljutni. A rövidtávú memóriában általában kevés információ szerepel, általában az, amire épp gondolunk, de korlátozott mennyiségben. A legtöbb ember 7 ( $\pm 2$ ) információt tud fejben tartani rövid ideig. Amikor a tanulók új ismerettel találkoznak a munkamemóriát használják, ahhoz, hogy sikeresen beépüljön az új ismeret, ahhoz az kell, hogy legyen kellő mennyiségű szabad memória. Minél komplexebb egy probléma, annál több figyelmet, memória egységet igényel.

Nagyon fontos a megfelelő tanulási környezet, hogy lehetőleg minél kevesebb inger érjen egy időben, így nagyobb mennyiségű memóriaegység jusson a feldolgozási folyamatra. Ez kiemelten fontos a tananyag fejlesztőknek, tankönyv, segédanyag összeállítóknak, nem kell, hogy színes szagos legyen, hanem, inkább arra kell törekedni, hogy a szereplő illusztrációk, ábrák valóban szorosan kapcsolódjanak a tartalomhoz, hogy a megértést elősegítsék, más esetben, csak feleslegesen foglalja a helyet a munkamemóriában. Az információ feldolgozása, és beépülése sikertelen, amikor a munkamemória terhelését meghaladja a tanulási feladat. A siker érdekében figyelembe kell venni a munkamemória korlátait, így a tananyagot, feladatot érdemes részekre bontani. Kirschner (2002) szerint az ember kognitív szerkezete memóriából és sémákból áll, ami egybehangzik Sweller (1998, 1999, Sweller et al. 2019) eredményeivel. A kognitív struktúra sémákon alapszik, amely fokozatosan épül, bővül. A sémák összekapcsolt elemekből és információkból állnak egy egységet alkotva, mely lehetővé teszi a munkamemória egy részének felszabadulását. A meglévő séma alkalmazása csökkenti a munkamemóriában az elfoglalt helyet, így komplexebb feladatok, problémák könnyebben és gyorsabban oldhatók meg. Az új ismeret a munkamemóriából a hosszútávúba úgy kerül át, ha elég hosszú ideig tartjuk a munkamemóriában. Ez az elmélet visszatükröződik Pólya (1954) problémamegoldási stratégiájában is.

Célunk volt, olyan megközelítések fejlesztése, gyakorlati alkalmazása és tesztelése, mely szakít a bevett hagyományos módszerekkel, ezzel szemben a problémamegoldást és a sémaépítést helyezi a középpontba. Az alkalmazott high-mathability<sup>7</sup> oktatási módszerek támogatják a tanulók gondolkodásmódjának fejlesztését, sémák építését és felidézését, valamint a problémák közötti

---

<sup>7</sup> A metakognitív számítógépes problémamegoldási megközelítések osztályozásra kerültek, a mesterséges és a természetes kognitív képességek matematika szempontjából való vizsgálata által, mely szerint a koncepció-alapú módszerek – amilyen a Pólya-féle módszer is –

kapcsolatok, összefüggések felismerését, valamint a hosszútávú tudás kialakítását (Csapó et al. 2020, Kahneman 2011, Sweller 1988, Sweller et al. 2011).

### 5.3. Didaktika és a problémamegoldás

A didaktika a pedagógia tudománya, amely azzal foglalkozik, hogy kinek, mit és hogyan tanítson (Báthory 1997; Nagy 1981; Székely 1979). A didaktika tárgyának/kérdéseinek a megválaszolásával kapjuk meg a hatékony tanítási, oktatási folyamatok ismérveit, a pedagógusok elvárt személyiségjegyeit és képességeit. A *kinek* és a *mit* kérdések szorosan kapcsolódnak egymáshoz, mivel kiemelten fontos szempont, hogy a tanulók életkori és egyéni sajátosságait is figyelembe vegyük az oktatási folyamat során. A *hogyan* kérdésre az alábbi didaktikai feladatok adnak választ, melyek elengedhetetlen pillérei az oktatási folyamatnak:

1. a figyelem felkeltése,
2. probléma megfogalmazása,
3. előismeretek felidézése,
4. az új ismeretek nyújtása,
5. a tények, jelenségek sokoldalú elemzése,
6. fogalomalkotás, következtetés absztrakciók,
7. rendszerezés és rögzítés,
8. alkalmazás,
9. ellenőrzés és értékelés

Kutatócsoportunk a módszerek kialakítása során törekedett ezeknek a didaktikai feladatoknak az átfogó lefedésére. A tanulók motivációjának elősegítése, fenntartása fontos szerepet játszik a tananyag elsajátíthatóságának szempontjából. Kozéki (1980) vizsgálatai alapján az értelmi tényezők csak 50%-

---

„matematikussága” a legmagasabb (5), míg a TAEW próbálgatásra épülőké a legalacsonyabb (1) (Baranyi és Gilányi 2013; Borus és Gilányi 2013).

ban befolyásolják a tananyag elsajátításának sikerességét, a további százalékokat a motiváció köré csoportosuló folyamatok határozzák meg. Kozéki iskolai motivációt mérő kérdőívével Kerekes (2013) is vizsgálatokat végzett a magyar tanulók körében, ahol Kozéki eredményével összehangban lévő következtetésekre jutott.

A tanulók figyelmét autentikus, érdeklődési körükbe tartozó, korának és tantárgyközi kapcsolatainak megfelelő, közös élményekhez kapcsolódó tartalmakkal keltjük fel (1). Ezen tartalmak elérésének, elemzésének és feldolgozásának motivációs faktora jelentős. A következő lépésben történik meg a probléma felvetése, amit a tanulóknak meg kell oldani, így a tanulók megismerik a célt, amit el szeretnének érni (2). Az alkalmazott oktatási módszerek a problémamegoldási stratégiák lépésein alapulnak, amelynek első fázisa a probléma megértése, elemzése (5), ehhez a tanulók felhasználják előzetes ismereteiket (3). Ezek alatt a munkafolyamatok alatt a pedagógus egyszerű kérdéseket feltevé irányítja a tanulók gondolkodását, mindemellett teret enged a kreatív ötletek születésének. A tanulók nem érzik feszélyezve magukat, mert ötleteiket természetes nyelven megfogalmazva oszthatják meg a többiekkel. Következő lépés a tervezés, a megoldási lehetőségek számbavétele, ehhez szükség van mind előzetes, mind új ismeretekre (4). Ezt követi a kiválasztott megoldási stratégia, a megoldáshoz szükséges eszközök alkalmazása (8). Negyedik lépés az ellenőrzés, a diszkurzus, ahol a tanulók megbizonyosodnak arról, hogy a megoldás elfogadható, ésszerű, következetes (9)(6). Amennyiben nem, megvizsgálják a megoldási stratégiát, az alkalmazott eszközöket és keresik a hibákat. Ezen lépések sorozata teszi lehetővé a sémákban való gondolkodást.

A kutatócsoportunk mindezek mellett, a fejlesztett és az alkalmazott módszerek hatékony oktatási alappilléreinek a rendszerezést és a rögzítést tartja.

“A rendszerezés és rögzítés elsődleges célja egy magasabb szintű, átfogóbb tudás létrejöttének elősegítése, ebben a fő hangsúly az általánosításon, az összefüggések keresésén és felismerésén, valamint a szintetizáláson van.” (Tóth 2015)

Az általunk kifejlesztett, vizsgált és alkalmazott módszerek a témakörök ismereteit szorosan kapcsolódva, egymásra épülve adják át, mellyel a tanulók az új ismeretet egy már meglévőhöz tudják kapcsolni (8), így a megértési folyamat gyorsabb. A korábbi ismeretek felhasználása, ismétlése egyúttal rögzítés is.

A magyar közoktatás általában ott hibázik, hogy nem alapozza meg a feladatokat, nincs célkitűzés, sem elemzés, hanem az alkalmazásra fókuszál, azt követően pedig nem alakítja ki az igényt az eredmény helyességének ellenőrzésére sem.

## **6. Kutatócsoport által fejlesztett és alkalmazott módszerek**

A Debreceni Egyetem Informatikai Karának docense, Csernoch Mária a programozás mintájára, a táblázatkezelés témakört problémamegoldás felől közelítette meg, alapul véve Pólya György problémamegoldási stratégiáját, kihasználva a táblázatkezelők saját nyelvét és szintaktikáját. Így született meg közel 20 éve a Sprego módszertan, amely az elmúlt 10 évben a kutatócsoport tagjainak, közös felméréseinek, oktatási tapasztalatainak köszönhetően továbbfejlődött, így a felsőoktatáson kívül, a közoktatásban, az általános iskolától kezdődően alkalmazhatóvá vált. A Sprego módszer pozitív tapasztalatait látva, a kutatócsoport hangsúlyt fektetett további informatikai alkalmazói területekre, amelyek tanítása során a felületalapú megközelítés vált általánossá,

ezáltal elavulttá. A Sprego módszertan továbbfejlesztése, segédeszközök készítése, alkalmazása és hatékonyságvizsgálata mellett a következő témakörök tanításához fejlesztettünk problémamegoldáson, sémaépítésen alapuló módszereket: fájlkezelés (Fájlkonverzió - Webtable-datatable Conversion), szövegkezelés (Hibafelismerési Modell - Error Recognition Model). Azon túl, hogy a módszerek alapja, módszere megegyezik, a tantárgyközi kapcsolatok és a tudástranszfer szorosan egybefűzi az említett 3 témakört.

### **6.1. Fájlkezelés**

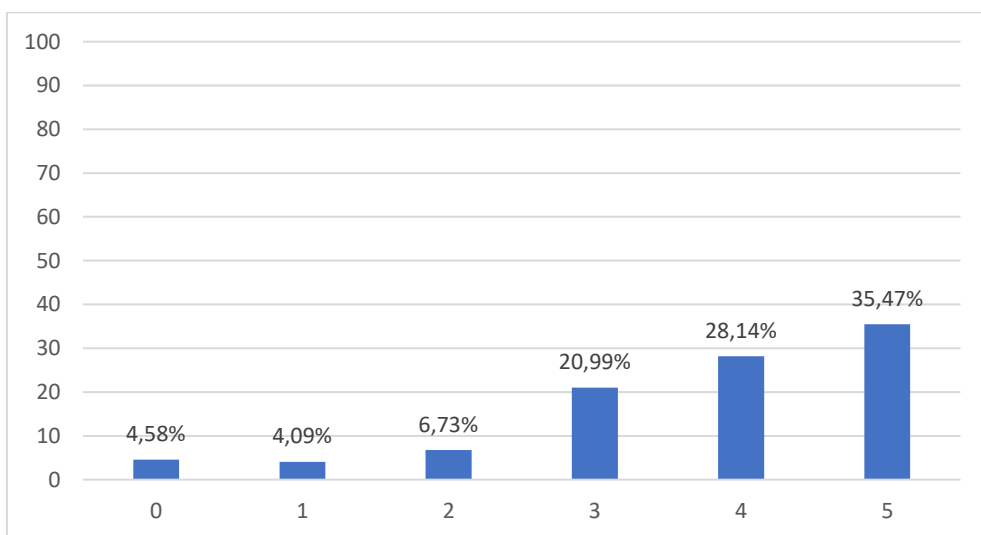
Az informatikaoktatás során az egyik legelső témakör, amivel a tanulók találkoznak az a fájlkezelés. Az elmúlt években azt tapasztaltuk, hogy egyre kevesebb időt fordítottak erre a témakörre az informatikatanárok, egyrészt az alacsony óraszám miatt (OFI 2012b), másrészt a digitális bennszülöttek mindennapi felhasználói a digitális eszközöknek, így ezeket az ismereteket már elsajátították (Kirschner & De Bruyckere 2017; Prensky 2001).

A digitális bennszülöttekkel kapcsolatos tévhitet tovább erősíti a 2018-as mini kompetencia teszt eredménye. A mini kompetencia teszt a Debreceni Egyetem hallgatójának, Nagy Tímea Katalinnak, egyben a kutatócsoport tagjának felmérése, amit 2018-ban bonyolított le 93 közoktatási intézmény részvételével. A felmérésben 7-10. évfolyamos tanulók vettek részt, összesen 8880 fő. Saját vizsgálataim elsősorban az első éves középiskolai tanulók hozott tudására irányult, így a mini kompetencia teszt eredményei közül elsősorban a vizsgálatban résztvevő 8880 tanuló közül 1562 fő 7. évfolyamos és 1643 fő 8. évfolyamos tanulók válaszait vettem alapul. A felmérésben a tanulók általános, statisztikai és demográfiai adatain túl, informatikaoktatásra vonatkozó kérdésekkel, valamint feladatokkal találkoztak. A tanulók önértékelés által határozhatták meg, hogy az informatika egyes témaköreiben milyen szinten jártasak egy 0-5-ig tartó skálán, melyek között szerepelt a fájlkezelés is (9. ábra).

G11. Hogyan értékelnéd ismereteidet az alábbi témakörökben? (0=egyáltalán nem tudom, 5=nagyon jól tudom)

	Mennyire ismered a témakört?						Iskolában tanultad?	
	Karikázd be a megfelelő számot!						tanultam	nem tanultam
fájlkezelés	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
szövegkezelés	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
táblázatkezelés	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
adatbáziskezelés	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
algoritmizálás, programozás	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
források kezelése, hitelessége	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>

9. ábra Mini kompetencia teszt önértékelés része, mely tartalmazza a fájlkezelés témakört is.



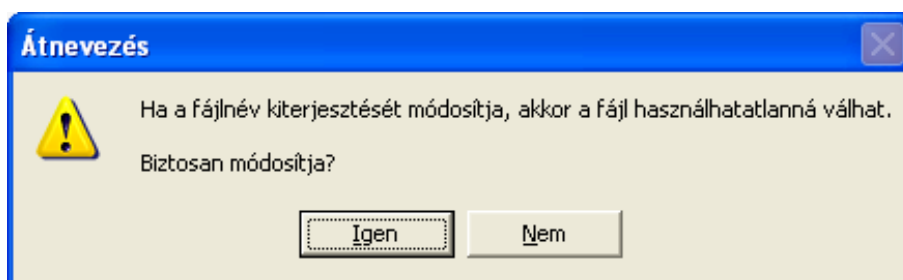
10. ábra A mini kompetencia teszt fájlkezelés témakörére adott 7-8. évfolyamos tanulók önértékelése 0-5-ig terjedő skálán megjelenítve, ahol az 5, a legmagasabb jártassági szintet jelöli (Sebestyén 2020).

Az önértékelés során a tanulók 79,06%-a választotta a 3-4-5 lehetőségeket (jó, nagyon jó, kitűnő), a tanulók harmada (32.64%) választotta a legmagasabb (5) lehetőséget. A mini kompetencia teszten a tanulóknak lehetőségük volt jelezni, hogy mely témaköröket tanulták iskolában. A tanulók 14,72%-a választotta azt, hogy nem tanulta az iskolában a fájlkezelést. A válaszaik alapján azok, akik nem tanultak fájlkezelést iskolai keretek között, hasonlóan magabiztosak, mint azok, akik tanultak. A válaszadók 40,04%-a a fájlkezelés téma-

kör 4-5-ös jártassági szintjét jelölte, ami azt mutatja, hogy a mai fiatalok ismerik és tudják alkalmazni a fájlműveleteket. Ez az eredmény látszólag alátámasztja és igazolja azt, hogy miért szorult háttérbe a fájlkezelés oktatása.

A témakör elhanyagoltságát és a módszer fontosságát jól szemlélteti az első éves középiskolai tanulók fájlkezelés ismereteit, tudásszerkezetét feltérképező vizsgálatunk eredménye (Sebestyén 2020). A felmérésben két iskola tanulói vettek részt, összesen 109 fő. A felmérésben 6 feladat (1. számú melléklet) megoldásával vizsgáltuk a tanulók algoritmizálási képességét, számítógépes gondolkodását, tudatos eszközhasználatát tudástranzfer elemeken keresztül.

Az első feladat (11. ábra) egy Windows-hibaüzenet értelmezése volt, ahol a tanulók megadott válaszlehetőségek közül választhatták ki a helyeset. A feladat megengedte több válasz jelölését, azonban csak egy helyes opció szerepelt a feltüntetett lehetőségek között.



11. ábra A fájlkezelés feladatsor első feladata, ahol a kiterjesztés módosítása során kapott hibaüzenetet kellett értelmezniük a tanulóknak.

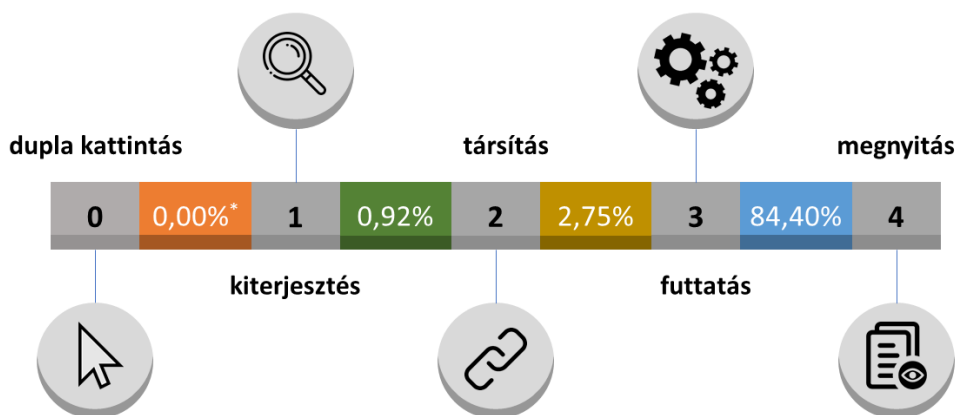
A tanulók a 3. táblázatban szereplő lehetőségek közül választhattak az első feladat megoldása során. A tanulók többsége, 45,87%-a jelölte az 5. helytelen válaszlehetőséget, ezt követte 37,61%-kal, majd 36,70%-kal a 3., valamint a 2. állítás. Meglepő módon a helyes megoldást, a 4. válaszlehetőséget önállóan, csupán a tanulók 15,60%-a jelölte, amely a legalacsonyabb mind közül.

<b>srsz.</b>	<b>állítások</b>	<b>%</b>	<b>pref.</b>
1.	Ha a kiterjesztés változik a gép képtelen lesz felismerni a fájlt.	29,36%	4.
2.	Kiterjesztés módosításnál adatok elveszhetnek.	36,70%	3.
3.	Megváltozik a társítás és a fájl használhatatlan lesz.	37,61%	2.
4.	Megváltozik a társítás, de a fájl továbbra is használható lesz.	15,60%	6.
5.	Ha rossz formátumra írjuk át a fájlt, akkor az megsérül és használhatatlan lesz.	45,87%	1.
6.	Az új kiterjesztés más kódolása miatt a tartalom a felhasználó által olvashatatlan lesz.	28,44%	5.

3. táblázat A fájlkezelés felmérés 1. feladatának megadott válasz lehetőségei, a válaszok jelölése %-ban kifejezve, valamint a válaszok preferencia sorrendje.

A felmérés feladataiban a tudástranszfer elemek ismétlődve szerepelnek különböző megközelítést használva. A kiterjesztéshez kapcsolódó ismeretek megjelennek az 2., 5. és a 6. feladatban is.

A tanulók 2. feladatként egy nyitott kérdést kaptak, ahol saját szavaikkal kellett megfogalmazniuk, hogy „Mi történik, ha duplán kattintunk egy dokumentumfájltra?”. A helyes megoldáshoz a tanulóknak ismerniük kell a dokumentumfájl fogalmát. Egyetlen tanuló sem oldotta meg hibátlanul a feladatot.



12. ábra A fájlkezelés felmérés 2. feladatának megoldásában szereplő 4-lépéses folyamat, valamint, hogy a tanulók hány %-a tudta azonosítani az egyes lépéseket. (\*a tanulók 11,93%-a ignorálta a feladatmegoldását)

A tanulók sok esetben egy-egy szóval, a megnyitással válaszoltak, ami csupán csak az utolsó fázisa a 4-lépéses folyamatnak. Ez megerősíti a tanulók első feladatban elért gyenge eredményét, amely azt mutatja, hogy a tanulók nincsenek tisztában az informatikai kifejezésekkel, fogalmakkal, úgymint kiterjesztés, társítás, dokumentumfájl (12. ábra).

A következő, harmadik feladatban a tanulók a fájlpushoz kapcsolódó tudásukról adhattak számot, amely ismeret megjelenik részben az 1. – az egyik leggyakoribb tévhit, hogy a kiterjesztés átírásával, beírásával megváltozik a fájl típus – és 6. feladatban is. Az első feladathoz hasonlóan a tanulók megadott válaszok közül, összesen 10, jelölhették meg a helyes választ arra a kérdésre, hogy „Hogyan tudnál egy táblázatkezelő dokumentumot (.xlsx vagy .ods) szövegfüggővé (.csv vagy .txt) alakítani? (egyet jelölhetsz meg)”. A feladatban egy helyes megoldás szerepelt, ennyit is engedett meg a feladatleírás, ennek ellenére a tanulók magas százaléka ezt az utasítást nem vette figyelembe. Összesen a tanulók 71,68%-a jelölte a „Mentés másként, kiválasztjuk az új típust.” helyes válaszopciót, ahol a tanulók közel fele más lehetőséget is helyesnek vélt. A tanulók 35,78%-a volt az, akik önállóan jelölték ezt a megoldást és szereztek pontot a feladatban. A helyes válasz volt a legtöbb tanuló által választott megoldás, ezt követte a *konverzió*, a *kiterjesztés átírása* és a *Mentés másként, átírjuk a kiterjesztést* lehetőség.

A negyedik feladat a kivágás operátorra kérdez rá, hogy mi történik, amikor ezt a műveletet végrehajtjuk (13. ábra). A tanulók 79,82%-a jelölte a vágólapra kerül opciót, önállóan csupán a tanulók fele, 52,29%-a. A helyes válasz mellett elsősorban, a tanulók 17,43%-a jelölte a *másolat készül róla*, valamint 8,26% a *lomtárba kerül* lehetőséget. A tanulók válaszai alapján, úgy tűnik, hogy a kivágás és a beillesztés parancsok szorosan összefonódtak, valamint, hogy egyfajta törlési lehetőségként azonosítják.

Mi történik egy fájl kivágásakor? (többet is megjelölhetsz)

- |   |   |
|---|---|
| <input type="checkbox"/> Lomtárba kerül.                | <input type="checkbox"/> Vágólapra kerül.     |
| <input type="checkbox"/> Véglegesen törlődik.           | <input type="checkbox"/> Nem történik semmi.  |
| <input type="checkbox"/> Megjelenik egy másik mappában. | <input type="checkbox"/> Másolat készül róla. |

13. ábra A fájlkezelés felmérés negyedik feladata, a kivágás operátor.

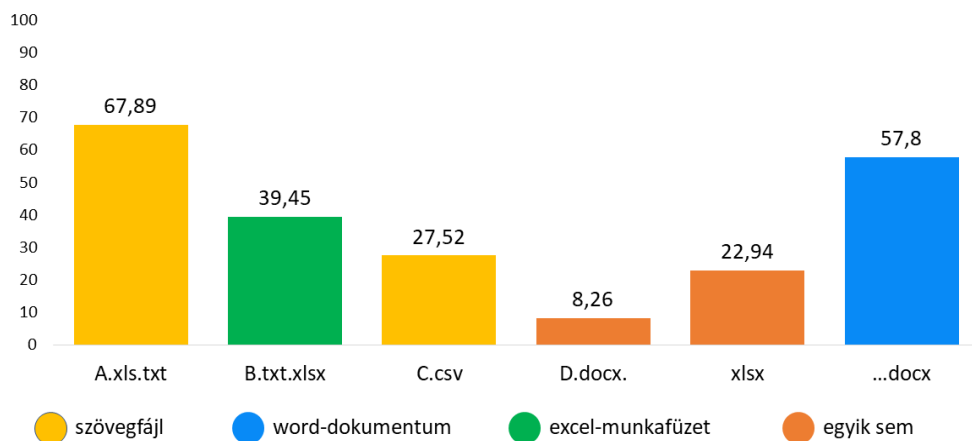
Az utolsó előtti, azaz az ötödik feladatban a tanulóknak a fájl neve és kiterjesztése alapján kellett eldönteniük, hogy milyen típusú a fájl (14. ábra). Attól függetlenül, hogy a feladatleírás megengedett több válaszlehetőséget, minden esetben csak egy jó válasz szerepelt. A megoldáshoz szükséges a fájl tulajdonságainak ismerete. Fontos kiemelni, hogy a fájlnevek nem a kutatócsoport értelmetlen kitalációja, hanem tanulók korábbi órai- és házi feladatai során elkészített dokumentumok nevei. A látszólag két kiterjesztéssel rendelkező fájlok tömeges megjelenése elsősorban a Microsoft Office 2020-as verziójához köthető, ahol a nem alapértelmezett kiterjesztésű fájlok megnyitása – .txt fájl Wordben, Excelben történő megnyitása –, majd mentése során a korábbi kiterjesztést fájlnevként megőrzi és a gyakorlatlan felhasználó ezen nem módosít. A másik lehetőség pedig a tanuló más típusként szeretné elmenteni a dokumentumot, azonban a fájltypust nem kiválasztja a legördülő listából, hanem a fájlnev után begépelni a kívánt kiterjesztést.

Válaszd ki, hogy milyen típusúak az alábbi fájlok! (többet is megjelölhetsz)

	szövegfájl	<u>word</u> -dokumentum	<u>excel</u> -munkafüzet	egyik sem
A.xls.txt	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
B.txt.xlsx	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
C.csv	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
D.docx.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<u>xlsx</u>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<u>...docx</u>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

14. ábra A fájlkezelés felmérés ötödik feladata, ahol a fájlok neve és kiterjesztése alapján meg kellett határozni a fájl típusát.

A tanulók elsősorban az *A.xls.txt* fájl típusát határozták meg helyesen, összesen 67,89% választotta ki a jó típust (15. ábra). Ezt a legutolsó *...docx* követi 57,80%-kal. A legkisebb sikerességgel a *D.docx*. (8,26%) és az *xlsx* (22,94%) típusát határozták meg (15. ábra) a tanulók. Egyetlen tanuló sem volt, aki az összes fájl típust helyesen határozta meg. A tanulók gyenge teljesítményének okát abban látjuk, hogy a tanórákon a Windows Intéző használata az elterjedt, ahol az alapértelmezett beállítás szerint a kiterjesztés rejtett. Mindemellett úgy véljük, és ezt a tankönyvi, valamint ECDL feladatok is alátámasztják, hogy a tanulók a tanórákon elsősorban előkészített fájlokkal, valamint a használt alkalmazások alapértelmezett kiterjesztéseivel dolgoznak.



15. ábra Az ötödik feladatban szereplő fájlnevek és helyes fájl típusai, valamint a helyes tanulói válaszok százalékban megadva.

Az utolsó feladat egy Igaz/Hamis feladat volt, amelyben a korábbi 1-5. feladatokban szereplő tudáselemek ismételten megjelentek, valamint a feladaton belül is ismétlődtek. Az állítások öt tudáselem köré csoportosíthatók, attól függően, hogy a tanulóknak milyen háttérismerettel kell rendelkezniük az állítás értelmezéséhez és helyes megoldásához (4. táblázat). A feladatokban elért önálló %-os eredmények a 4. táblázatban szerepelnek.

Megvizsgáltuk a tanulók válaszait, hogy mennyire voltak következetesek, milyen sikerességgel oldották meg a hasonló ismeretet igénylő feladatokat,

azonban véletlenszerű mintázatot kaptunk. A mintázat hiánya arra enged következtetni, hogy a tanulók ismeretei hiányosak és/vagy töredezetek, így első-sorban tippeltek a feladat megoldása során.

	1	2	3	4	5	6	7	8	9	10	11	12
kiterjesztés	●	●	●					●				
társítással/társítás nélkül	●	●	●	●	●	●	●					
fájltípus						●	●	●	●	●	●	●
fájl szerkesztés, mentés											●	●
fájl megnyitás				●	●	●			●	●		
	45.87	82.57	22.02	49.54	44.95	29.36	41.28	30.28	31.19	43.12	22.94	61.47
eredmények %-ban												

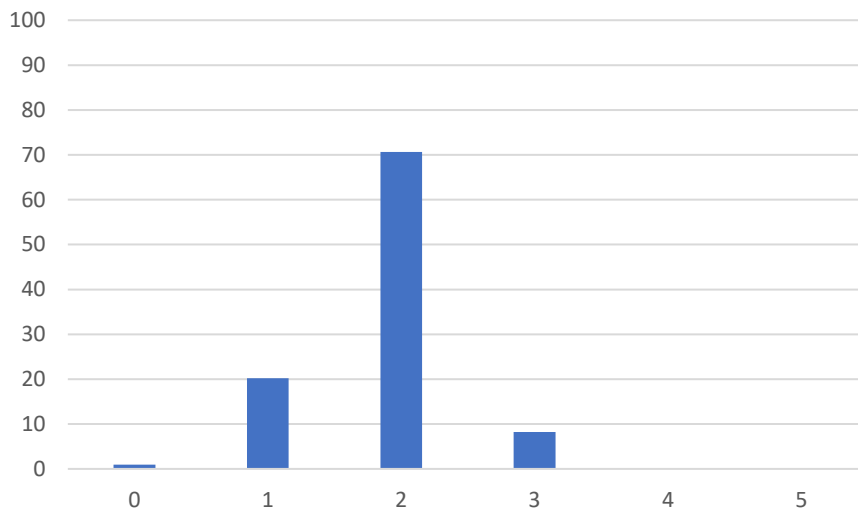
4. táblázat A fájlkezelés felmérés 6. feladatában megjelenő tudáselemek (első oszlop); a feladat 12 állítása és a helyes megoldásukhoz szükséges tudáselemek fekete ponttal kerültek jelölésre. A feladatokban elért tanulói eredmények %-ban megadva található az utolsóelőtti sorban.

A fájlkezelés felmérés 9. évfolyamos tanulói az összes feladatot együttvéve 32.34%-ra teljesítették, ami meglepően alacsony a 3 évnyi iskolai informatikaoktatás mellett (5. táblázat). A 6. feladatban a mintázat hiánya (4. táblázat), valamint az összeredmény megerősíti a [T3] tézis felvetést, mely szerint az első éves középiskolai tanulók nem, vagy alig tudnak támaszkodni az általános iskolából hozott ismereteikre.

	helyes válaszok (%)
F1	4,59
F2	22,02
F3	35,78
F4	52,29
F5	37,31
F6	42,05
összesen	32,34

5. táblázat A fájlkezelés teszt eredményei feladatonként. (Sebestyén 2020)

A tanulók egyéni eredményeit megfeleltettük egy 0-5-ig terjedő skálának (a százalékos eredmény visszaosztásával), hogy összehasonlítható legyen a mini kompetencia teszt önértékelési eredményeivel. Az így kapott eredményeket a 16. ábra jeleníti meg.



16. ábra A fájlkezelés teszt eredményei 0-5 közötti skálán megjelenítve.

A 16. ábra szerint egyetlen tanuló sem volt, akinek tudása 4-5 jártasságiszínten van (Sebestyén 2020). A tanulók tudása és önértékelése közötti különbség (10. ábra, 16. ábra) egyértelműen igazolja a [T4] tézist, mely szerint a tanulók az informatikai ismereteiket az önértékelés során túlértékelik. Ezek az eredmények sarkalltak minket a fájlkezelés tanításának fontosságának felhívására. A tanulóknak, akik a digitális bennszülött generáció tagjai, szükségük van informatikaórákra, legyen bármily „jelentéktelen” témakör, mint a fájlkezelés. A tanulók saját tapasztalataik alapján – próbálgatással – tanulták meg a számítógép és az okostelefonok használatát, melynek oka a késői (6. évfolyam) informatikaoktatás bevezetése. Az így megszerzett tudás az évek alatt tanulóként eltérő szinten bevésődött, így kiemelten fontos az ismeretek rendszerezése, pontosítása, az informatikai eszközök tudatos használatának kialakítása,

a számítógépes gondolkodás fejlesztése és a számítógéphasználathoz köthető terminológia elsajátítása.

### **6.1.1. Fájlkonverzió (WDC<sup>8</sup>)**

A módszer fejlesztése szorosan kapcsolódik a táblázatkezelés tanításához, ugyanis mindennapi példákon, autentikus források felhasználásával motiváltuk a tanulókat. Így ahelyett, hogy előkészítettük volna a táblázatokat, témákat, tartalmakat, a tanulók maguk, közvetlenül az internetről letöltve érték el az adatokat, sok esetben nem is táblázat formájában. A kutatócsoport ekkor ismerte fel, hogy az autentikus tartalmak letöltése, konvertálása – a weblapból hogyan lesz táblázat – a fájlkezelés eszközeit és néhány alkalmazói szoftver adatmanipulációs műveletének használatát igényli (Csernoch & Biró 2019, Csernoch & Dani 2017). Így magát a konverziót, konverziós technikát kezdtük használni a fájlkezelés eszközeinek ismertetésére, rendszerezésére és elmélyítésére.

A módszer high-mathability megközelítést alkalmaz, amely séma- és algoritmusépítésen alapul. Az oktatási folyamat elején a tanár felvet egy problémát: hogyan lehet az interneten található webtáblákat (vagy táblázatnak tűnő szerkezeteket) táblázattá alakítani. A probléma megértéséhez, megoldásához a tanulók matematikai előismereteit elő kell hívni. Értelmezniük kell, hogy mi a különbség a webtábla és az adattábla között, ezt a tanár rövid, célzott kérdésekkel irányítja, hogy a tanulók meg tudják fogalmazni a különbségeket, megértsék a problémát és megtalálják a megoldást. A tanulók nem kezdenek azonnal a számítógépen kattintgatni, próbálgatni, hanem attól eltávolodva gondolkodnak, ötletelnek, természetes nyelven fogalmazzák meg a problémát, a nehézségeket, akadályokat és próbálnak a megoldáshoz szükséges eszközöket,

---

<sup>8</sup> A módszer angol elnevezésének rövidítése: webtable-datatable conversion.

módszereket javasolni, amelyet a csoport közösen megvitat. A konverziós technika rámutat arra, hogy nem mindig egyértelmű mely eszközt kell alkalmazni a megoldás során. A webtáblák átalakítását a legtöbb tanuló és felhasználó valamely táblázatkezelői programban próbálná megoldani, ehelyett azonban sok esetben egy szövegszerkesztő sokkal célravezetőbb, melynek eszközeivel könnyen egy letisztult táblázat nyerhető ki. A konverzió lépései során számos fájlformátummal (html, docx, txt, xlsx) találkozhatnak a tanulók, amelyek esetében nem a társított programmal kell megnyitni a fájlokat. Ennek következtében a tanulók fogalmirendszere bővül és letisztul, valamint kreatív ötletek születését alapozza meg.

Az első néhány konverziós feladat során a tanulók segítségével a Redmenta felületén elérhető sorba rendező és párosító feladat áll, ahol a probléma megoldásához szükséges algoritmus lépései, valamint az elvégzendő műveletek találhatóak, melyek között szerepelnek a fájlműveletek is: megnyitás, mentés, mentés másként, létrehozás, bezárás stb. Ezek helyes sorrendje és párosítása támaszt nyújt az önálló feladatmegoldáshoz is. Nincs két egyforma konverzió, de a folyamat azonos sémára épül, ahol az első és az utolsó lépések megegyeznek, míg a közbelső mag alkotja a változó részt, ami új ötleteket, kreatív megoldásokat vár el a tanulóktól. 3-4 konverziós feladat megoldása fejleszti a tanulók gyors gondolkodását, sémaépítését és képes beépülni a hosszútávú memóriába. Kiemelten fontos, hogy a tanulók ne csak lépéseket kövessenek, mint az ECDL feladatokban (ECDL Foundation 2019, Papp & Csernoch 2018, Dancsó & Korom 2013), hanem fejlesszük a tanulók sémaépítési és -alkalmazási képességeit, amelyek lehetővé teszik a gyors- és lassú gondolkodás hatékony alkalmazását (Kahneman 2011). Célunk, hogy a tanulók ne csak leírást, receptkönyvet tudjanak követni a feladatmegoldások során, hanem aktívan vegyenek a valós világ problémáinak megoldásában. A kutatócsoport fontosnak tartja a tanulók megszólaltatását is, melyet a Pólyától származó kutatás alapú

tanulás (Pólya 1954) vagy ehhez a megközelítéshez szorosan köthető, napjainkban egyre népszerűbb coaching technikát alkalmazza (Chmielewska & Gilányi 2019). Az alkalmazott technika a hagyományos módszerekkel ellentétben megszólaltatja a tanulókat, mely támogatja a tanulók nyelvi és kommunikációs készségeinek fejlesztését, valamint az informatikához köthető terminológia helyes használatát.

## **6.2. Szövegkezelés**

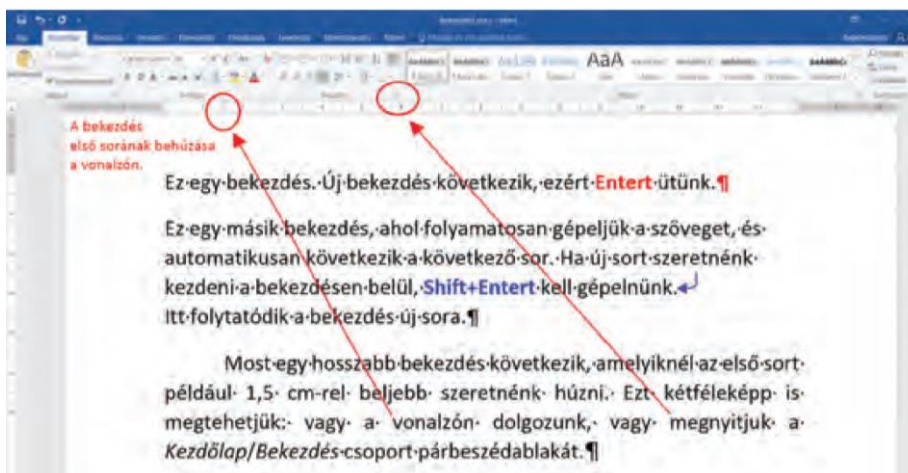
Ez az a témakör, amelynek az ismereteire mindenkinek szüksége van, de csak kevesen alkalmazzák jól (1. ábra, 2. ábra). A táblázatkezelői dokumentumokhoz hasonlóan az interneten található és letölthető szöveges dokumentumok nagy százaléka hibás, függetlenül a készítőjétől, ez lehet állami hivatal, oktatási intézmény, vállalat vagy magánszemély. A jól formázott dokumentum – legyen az nyomtatvány, önéletrajz, kísérőlevél – nem csak az informatikai tudásunk prezentálja, hanem, hogy mennyire vagyunk pontosak, precízek és igényesek a munkánkra. A helyesen alkalmazott eszközök pozitív képet adnak rólunk, vállalatunkról. Szövegkezelési ismeretek nélkül a felhasználó gyakran próbálgatással és barkácsolással készíti a szöveges dokumentumokat, ahol elsősorban az számít, hogy nyomtatásban hogyan néz ki, nem pedig az, hogy a dokumentum helyes-e (Ben-Ari 1999, Ben-Ari és Yeshno 2006, Csernoch 2009, 2010, 2011, Sebestyén et al. 2021). Az így készült dokumentumok módosítás során, a felhasználó eredeti szándékán túl, további extra formázást és gépelést igényelnek, ami időigényes. A leggyakoribb tévhit és kifogás a felhasználók részéről, hogy tudják helyesen alkalmazni a szövegkezelők eszközeit, például a térközt, de egyszerűbb az Entert nyomni, ezzel időt spórolnak meg. Ez az állítás azonban nem helytálló már 1 oldalnál hosszabb szöveg esetében. A közoktatás szereplői gyakran készítenek kiselőadásokat, gyűjtenek

adatot különböző témakörökhöz, házidolgozatokhoz vagy vesznek részt projektmunkákban. A felsőoktatásban pedig általános a beadandó dolgozat és a szakdolgozat. Ezek mindegyike több oldalas dokumentum, az el-/előkészítésük során nélkülözhetetlen a szöveges dokumentumok tervezése, létrehozása, szerkesztése és formázása. A leggyakoribb hibák és barkácsolások közé tartozik az Enterek ütése térköz helyett, vagy éppen minden sor végén Enter használata, ahogy régen az írógépen; a szóközzel való igazítás behúzás vagy tabulátor helyett. Ezek valóban nem látszódnak nyomtatásban, azonban időigényesek, hiszen 8-12 szóköz (2. ábra) lenyomása minden sorban tovább tart, mint bármilyen behúzás, tabulátor beállítása. Ezen kívül ott vannak azok az esetek, amelyek módosítás nélkül nyomtatásban is látszódnak, ilyen a kézi számozás automatikus helyett vagy a tartalomjegyzék. Az utólag beillesztett pontoknál/fejezeteknél az azokat követő minden pont/fejezet számát – és a hozzá tartozó oldal számát – módosítani kell, ami időigényes, hibázási lehetőség, javítás nélkül pedig helytelen adatokat (oldalszám, fejezetsorrend) közöl.

A közoktatásra továbbra is jellemző, ahogy minden nem programozási témakör esetében, hogy felületalapú megközelítést alkalmaz. A hozzá tartozó segédezközök, tankönyvek és feladatok elsősorban a szoftverben történő navigációt támogatják. Ezt a megközelítést a szoftvergyártó cégek is pártolják, melyet a „felhasználóbarát” szlogenbe zártak. Ennek következtében a tanulók nem képesek elsajátítani a szövegkezelést, a dokumentumtervezés lépéseit, a helyes formázás és tördelés ismerveit. A felületen történő navigáció során a hosszútávú memóriába csak egy-egy gomb, ikon, funkció helye rögzül, ezek helyes alkalmazása nem. A tanulók nem építenek sémákat, így sokkal több munkamemóriára van szükség a dokumentum készítéséhez, szerkesztéséhez, ami hibákat eredményez.

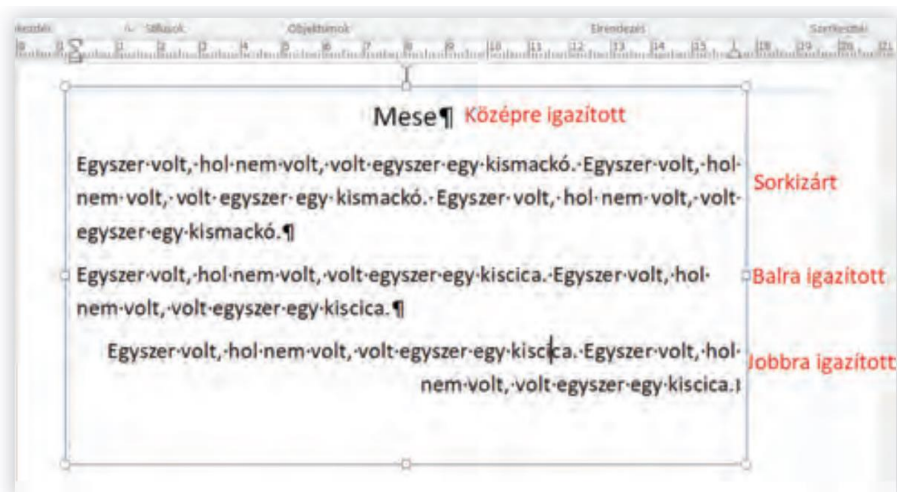
A közoktatásban elérhető hivatalos tankönyvekben, segédanyagokban szereplő feladatok többsége nem igazi feladat, nincs tartalom, hanem a formázandó szöveg egy-egy definíció, vagy csak egyszerűen „ez a sor legyen félkövér” (17. ábra) felirat.

Ahol ettől eltérő szöveg van, ott a tartalomtól független összevissza karakter- és bekezdésformázásokat hajtanak végre, vagy, ahogy a 18. ábra is mutatja, ugyanazt a mondatot megtöbbszörözve használja, és különböző módon formázza, ami nem illik a szöveghez és megbontja annak egységét, továbbá nem segíti a tanulók esztétikai érzékének fejlesztését. A tankönyvekből kiemelt képek és ábrák a tankönyvek online változatából származnak, így minőségük azzal megegyező (17. ábra, 18. ábra).



17. ábra A Nemzeti Tankönyvkiadó Informatika 7. című könyvének Szövegszerkesztés fejezetéből (Lakosné et al. 2019b)

Az Informatika 6. című tankönyvben (Lakosné et al. 2019a) a Kiadványszerkesztés fejezetben ugyanazon a szövegen szemléltet minden bekezdésformázást (18. ábra). A tankönyv illusztrációi elsősorban a felületen való navigációt szemlélteti képekkel, ahol gyakran egy példán keresztül mutat be mindent.



18. ábra Szövegszerkesztés mintafeladat és magyarázóképek az Informatika 6. tankönyvből. (Lakosné et al. 2019a)

Ez a megközelítés nem kelti fel a tanulók érdeklődését és nem is tudja fenntartani a figyelmüket. Azon túl, hogy a felületalapú megközelítéssel csupán egy adott verziójú szoftver használatának elsajátítását teszi lehetővé – ami a közoktatásban általában 5-10 évvel el van maradva a munkaerőpiacon használttól –, az átadott ismeretek nem biztosítanak hosszútávú tudást.

A természetes nyelvű szövegkezelést tovább nehezíti, hogy nincsenek hibakeresők, elérhető és megbízható automatizált ellenőrzők, mint a mesterséges nyelvek esetén a programozásnál. A szövegkezelés során egyetlen rendelkezésre álló eszköz van, mégpedig a helyesírásellenőrző, amelynek hatékonysága függ a nyelvtől, a helyes tördeléstől, tehát a felhasználó tudásától. Egyes szoftverekben található automatikus javítás, azonban ezek még kiforrotlanok, néha több kárt okoznak, mint amennyit használnak. A természetes nyelv összetettsége miatt a kutatócsoport indokoltnak látja, hogy a felhasználók tanulják meg a digitális szövegekben előforduló hibák felismerését, kezelését és elkerülését.

A szövegkezelés oktatása nehéz, hiszen általában valamely tantárgyhoz kapcsolják, Magyarországon az informatikához, azonban a természetes nyelvű

szövegekkel szemben támasztott követelmények sokfélék. Az informatikatanárok ezen követelményeknek, kompetenciáknak csak egy részét ismerik mélyrehatóan, így kiemelten fontos lenne a szövegkezelés oktatására csoportmunkaként tekinteni, ahol kiemelt szerepet játszik a magyar nyelvre szakosodott pedagógus. Az ő feladatuk a magyar nyelv szintaktikai és szemantikai szabályainak az oktatása.

### **6.2.1. Hibafelismerés és keresés (ERM<sup>9</sup>)**

A szövegkezelés oktatására a kutatócsoport a Hibafelismerési modellt fejlesztette ki, majd vizsgálta a módszer hatékonyságát a felületalapú, ECDL feladatokat előnyben részesítő hagyományos módszerek segítségével.

A legtöbb felhasználót nem tanítják meg, hogy hogyan kell dokumentumot tervezni, algoritmust készíteni a szövegkezelési folyamatokhoz és hogyan kell helyesen formázni. A tanulók nem ismerik azt sem, hogy a jólformázott szöveggel szemben milyen követelmények, elvárások támaszthatók. Ez az állapot tipikus példája a Dunning-Kruger effektusnak (Kruger & Dunning 1999).

A hibák felismerése, javítása kulcsszerepet játszik a helyes digitális dokumentum elkészítésében. A szöveg jólformázottnak tekinthető, ha megfelel a nyomtatott szöveg összes követelményének, tehát amit a nyomtatásban látunk, és ami mögötte van (digitális dokumentum), a módosításokkal szemben invariáns – a módosítás, kiegészítés nem von maga után extra felesleges formázásokat és gépeléseket. A kutatócsoport által fejlesztett és alkalmazott minden oktatási módszer során a képzési folyamat alatt valós problémákkal, szövegekkel, táblázatokkal dolgoznak a tanulók. A szövegkezelésoktatása során sem térünk el a problémamegoldás-alapú koncepciótól, azonban érdemes megváltoztatni a végrehajtási sorrendet a megoldási folyamat során. Előtérbe kerül a

---

<sup>9</sup> A módszer angol elnevezésének rövidítése: Error Recognition Modell.

debugging, a hibakeresés. Ez a megközelítés széleskörben elfogadott a programozásoktatásban, ahol a hibás programok, pszeudo kódok és modellek elemzésének segítségével sajátítják el a tanulók a programozási ismereteket (Gould 1975; Freiermuth et al. 2008; Bell & Newton 2013; Panko 2013; Jerinic 2014; Gander 2014). A hibák azonosításával, felismerésével sajátítják el azt a gondolkodásmódot, mely szükséges az optimális megoldáshoz szükséges eszközök kiválasztásához.

A hibakeresés és felismerés tanórai alkalmazása formabontó, hiszen egy-egy feladat értékelése, hibáinak keresése és javítása általában a tanár feladata. A hibafelismerés és keresés során ez a tanulókra hárul, amely egyrészt aktív részvételt követel a tanulóktól, másrészt egy mély figyelmi állapotot hoz létre, előtérbe helyezi és fejleszti a tanulók kritikus gondolkodását és autonómiáját. A természetes nyelvű szövegekben előforduló lehetséges hibák száma magas, ezért osztályozni kell őket. A hibák kategorizálása segít, mind a tanároknak, mind a tanulóknak a hibák egyértelmű azonosításában, valamint keretet ad. A hibák fő osztályai szabályrendszeren alapulnak, amelyek útmutatást adnak a hibák helyes felismerésében és besorolásában. Az osztályozáskor figyelembe vettük, hogy a digitális szövegnek meg kell felelnie a nyelvhelyességi és a szöveggel szemben támasztott követelményeknek, továbbá a megjelenítésre, elrendezésre) és formázásra vonatkozó szabályoknak (Jury 2004, 2006; Reynolds 2008; Virágvölgyi 2004). Ezen szabályok szerint a következő hibacsoportokat, típusokat különböztetünk meg: szintaktikai, szemantikai, tipográfiai, tördelési, formázási és stílus (Csernoch 2016, 2019; Csernoch & Biró 2015a). Az előforduló hibák felismerése, besorolása és javítása a tanulók észrevételein, ötletein alapulnak, amely támogatja és fejleszti a tanulók önálló problémamegoldási stratégiáit.

A digitális szöveg kezelése során döntő szerepe van a nem nyomtatódó karaktereknek, amelyek elsősorban elválasztó, határoló jelként funkcionálnak.

Ezek ismerete nélkülözhetetlen a digitális szövegek, a természetes nyelv megértéséhez és helyes alkalmazásához. A szövegszerkesztők képesek megjeleníteni a nyomtatásban nem látszódó karaktereket, azonban ezt a legtöbb felhasználó nem alkalmazza, mert zavarónak találja, azonban ezen karakterek megjelenítése lehetővé teszi a hibák észrevételét és korrekcióját a szerkesztési folyamat során. Az alkalmazói ismeretek oktatása során a korábban is említett, felületalapú, low-mathability, navigációközpontú módszerek terjedtek el, azonban a programozásoktatásban elsősorban a problémamegoldási stratégiák, algoritmusépítés áll a középpontban. Ennek alapja a Pólya György (1954) koncepcióalapú problémamegoldási stratégiája, amely a programozáson túl a matematika terén is hatékonynak mutatkozott, valamint az alkalmazói ismeretek oktatásában is pozitív eredményeket hozott.

A gyakorlatban a hibakeresés és felismerés – ERM – megközelítés alkalmazása során hiteles, adekvát szövegeket használunk a tanórákon, amelyeknek elsődleges forrása az internet. A fájl típusra való keresés több ezer dokumentumot eredményez magyar nyelven is, és ezek mindegyike szinte kivétel nélkül tartalmaz hibát. A nehézséget elsősorban az okozza, hogy a tanulók érdeklődésének és életkori sajátosságainak, demográfiai háttérüknek, valamint az informatika órák pedagógiai céljainak megfelelőek legyenek. Ugyan a keresési folyamat időigényes, azonban kiemelten fontos a hitelesség, a tanulók nem érzik úgy, hogy ezek a szövegek, anyagok kifejezetten oktatási célból készültek, ezáltal kapcsolatot teremt az iskola és a való világ között. A hibás forrásokat a kutatócsoport folyamatosan gyűjti, amelyek a különböző korosztályú és érdeklődésű tanulók oktatására alkalmasak, több mint 100 kipróbált dokumentum áll rendelkezésre a módszer iránt érdeklődő tanárok részére.

Az oktatási folyamat 3 szakaszra osztható, mely megegyezik a hatékonyságvizsgálat során alkalmazottal is. A szakaszokat a szerint különböztetjük

meg egymástól, hogy milyen mértékben van szükség informatikai eszköz használatára. Az első az unplugged, amelyben a tanulók nem használnak számítógépet, a második a semi-unplugged, ahol részben van szükség számítógépre, és az utolsó a plugged-in, ahol minden munkafolyamat a számítógépen történik (Sebestyén et al. 2021).

Az unplugged szakaszban a tanulók egy nyomtatott dokumentumot kapnak, amelyben nyomtatásban is látszó hibákat keresnek. Az így észrevett hibákat jelölik, rendre kék tollal, majd szöveges magyarázatot, indoklást írnak mellé, valamint meghatározzák a hibák típusát.

A középső szakaszban a tanulók a korábban kapott nyomtatott dokumentum digitális verzióját megnyitják számítógép segítségével és láthatóvá teszik a nyomtatásban nem látszó karaktereket is, amelyek további hibákra mutatnak rá. A tanulók tollat cserélnek, így az újonnan felfedezett hibákat pirossal jelölik az első szakaszban kapott nyomtatott dokumentumon.

Az utolsó, plugged-in szakaszban a tanulók kijavítják a digitális dokumentum hibáit és végül a szöveg helyes formázását végzik el.

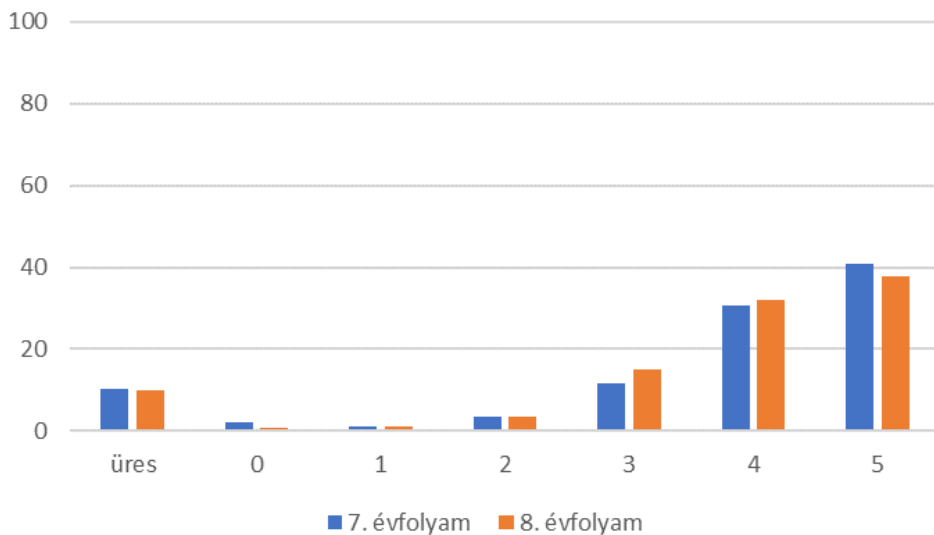
Az oktatási folyamat során a tanulók önállóan keresik a hibákat az első két szakaszban, azonban ezeket a tanulócsoport közösen ellenőrzi és keres további hibákat, amelyek felett elsikkadtak. A tanár elsősorban irányítja a tanulók figyelmét és rendszerezi a tudásukat. A módszer a hibák felismerésén alapul, de ugyanilyen fontos, hogy a diákok meg is tudják fogalmazni természetes nyelven a hiba okát, és hogy hogyan valósítható meg helyesen. A magyarázat segíti a megértési folyamatot és rögzíti a helyes megoldásokat, módszereket, valamint támogatja a helyes terminológia használatát.

A Fájlkonverzió módszerhez hasonlóan a szövegkezelés esetében is összehasonlításra került 7., valamint 9. évfolyamos (10. táblázat) tanulók általános iskolából hozott tudása (Sebestyén et al. 2021) a mini kompetencia felmérés önértékelésén (0-5 Likert-skála segítségével) alapuló eredményeivel. Ezek

a tanulók a kerettanterv szerint 6-8. évfolyamon tanultak szövegkezelést, így a mini kompetencia felmérés 8880 résztvevője közül 1562 fő 7. évfolyamos és 1634 fő 8. évfolyamos tanuló eredményeit vetettük figyelembe. A mini kompetencia felmérés a tanév végén zajlott le, így a felmérésben részt vevő tanulók már elsajátították az adott évfolyamra ütemezett szövegszerkesztési ismereteket.

A tanulók hozott tudásának feltérképezésének az Hibafelismerés és keresés (ERM) módszer hatékonyvizsgálata adott keretet. A tanulók az oktatási folyamat előtt egy előteszten vettek részt, melynek eredménye tükrözi a témakörben szerzett jártasságukat. Az ERM felmérésben résztvevő tanulók száma 153 fő, akik az előteszt feladatát (3. számú melléklet) a korábban ismertetett három szakasz mentén oldották meg – unplugged, semi-unplugged és plugged-in. Az eredmények összehasonlításakor a tanulók első két szakaszban elért eredményeit elemeztük.

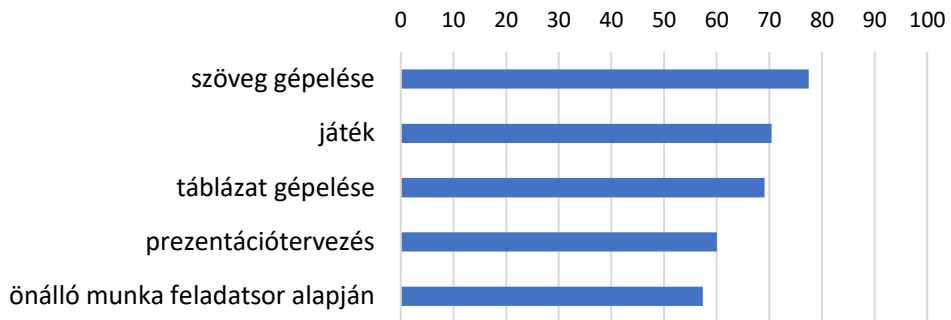
A fájlkezelés után a szövegkezelés volt az a témakör, amelyben a legmagabiztosabbnak érzik magukat a tanulók a mini kompetencia felmérése szerint. A különböző évfolyamok (7-8.) önértékelési eredményei szorosan együtt mozognak, nincs szignifikáns eltérés a tanulók önértékelése között (19. ábra), pedig 1 év informatikaoktatás választja el őket egymástól. Ez vajon tipikusan a Dunning-Kruger-hatás (Kruger & Dunning 1999), ezért olyan magabiztosak a 7. évfolyamosok, vagy pedig ennyire nem számít 1 év informatikaoktatás?



19. ábra Mini kompetencia felmérés önértékelésének eredménye szövegkezelés témakörében évfolyamonként.

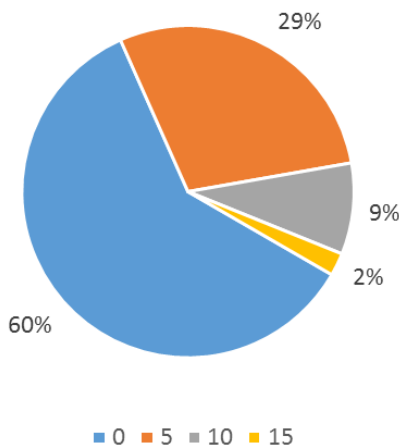
A válasz mindkettő vagy éppen egyik sem. A 19. ábra megmutatja, hogy a legmagasabb jártassági szintet az alacsonyabb évfolyam tanulói magasabb százalékban jelölték, igaz a különbség nem szignifikáns (7. évfolyam 40,85%; 8. évfolyam: 37,80%). Emellett fontos kiemelni, hogy a fájlkezelés esetében is több tanuló jelölte, hogy nem tanulta iskolában a témakört, ez a szövegkezelés esetében sincs másképp. A tanulók 5,90%-a jelölte a nem tanulta válaszlehetőséget. A fájlkezelés esetében értettük (de nem fogadtuk el) a magyarázatot a háttérben, hogy a digitális bennszülött generáció tisztában van a fájlműveletekkel, tudja kezelni az operációs rendszert és annak felületét, így annak oktatása elhagyható. A szövegkezelés azonban más terület. Hány fiatal használ szabadidejében szövegszerkesztőt? A szám nullához közelít, így a korábbi magyarázat nem állná meg a helyét, hogy miért hagyná el az informatikatanár a témakört. Azonban mégis előfordulhat. Erre szintén választ adhat a mini kompetencia mérés, ahol a tanulók jelölték azt is, hogy milyen tevékenységek jelennek meg a tanórákon (20. ábra). Az első helyen a szöveg gépelése jelenik

meg, amely az egyik legfeleltesebb és legidőigényesebb tevékenység az informatikaórákon. A tanulók eltérő sebességgel gépelnek, így a csoport együtt haladása nem biztosított, ezen felül nem fejleszt egyetlen olyan képességet sem, amely megjelenik a nevelési és oktatási célok között. Ezt követi a játék és a táblázat gépelése, amelyek szintén az előbbi kategóriába tartoznak.

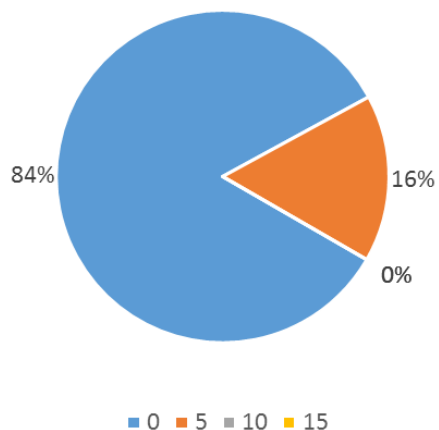


20. ábra A mini kompetencia felmérésben jelölt első öt, azaz legjellemzőbb tevékenység listája a 7.-8. évfolyamos tanulók válaszai alapján.

Ezzel az eredménnyel összhangban van egy korábbi kutatás eredménye (Sebestyén 2014), ahol 41 általános- és középiskolai informatikatanár vett részt egy kérdőíves felmérésben.



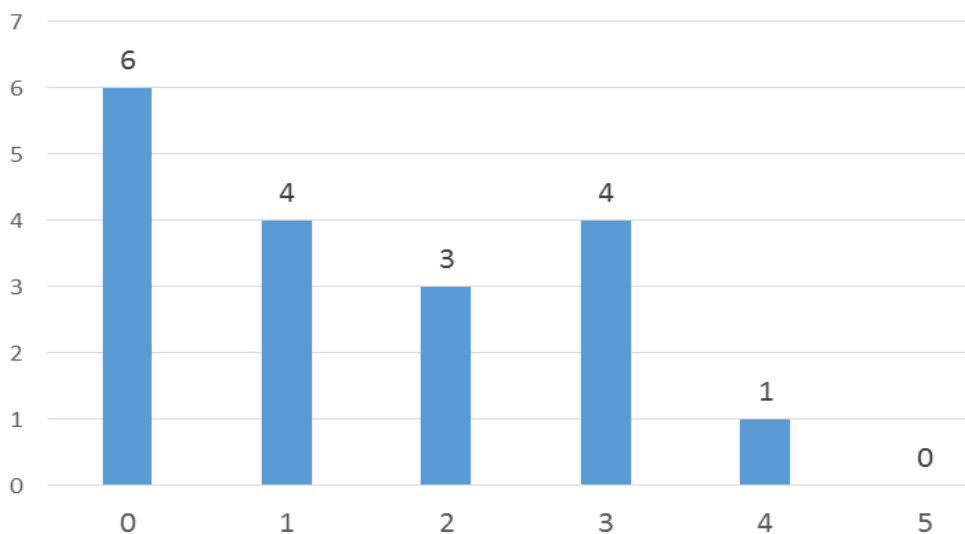
21. ábra Az informatikatanári kérdőív összesítése alapján az informatika tanórákon megjelenő játék, szabadfoglalkozás aránya percben kifejezve.



22. ábra Az informatikatanári kérdőív összesítése alapján a másik szak tanóráin megjelenő játék, szabadfoglalkozás aránya percben kifejezve.

A válaszadók 40%-a biztosít lehetőséget az informatika tanórákon játékra különböző mértékben (21. ábra). A válaszaik több szempontból ellentmondásosak, egyrészt azok a tanárok, akik 5-15 percet szánnak játékra, a leginkább elégedetlenek a Kerettanterv (OFI 2012b) által biztosított óraszámokkal (23. ábra). Másrészt a másik szakjuk esetében – ami jellemzően matematikai, fizika – kevesen (16%) és elenyésző mértékben, maximum 5 percet fordítanak játékra (22. ábra) (Sebestyén 2014).

Ezek megmagyarázzák, hogy hogyan fordulhat elő, hogy a tanulók nem tanulták a szövegkezelést a tanórákon és egyben igazolják, hogy miért olyan alacsony a témában szerzett jártasságuk.



23. ábra Informatika tanárok elégedettsége a Kerettantervben (OFI 2012b) foglalt óraszámokkal, akik engedélyezik a játékot a tanórán. (0 – elégedetlen; 5 – elégedett)

A Hibafelismerés módszerének hatékonyságvizsgálatának előtesztjén a tanulók hozott tudását mértük fel, az eredményeket évfolyam szerint csoportosítottuk (Sebestyén et al. 2021). A vizsgálat során a tanulók hibákat kerestek nyomtatott, majd digitális dokumentumban, amelyeket egy lapon jelöltek és magyaráztak, valamint utolsó lépésben kijavították a digitális dokumentumot.

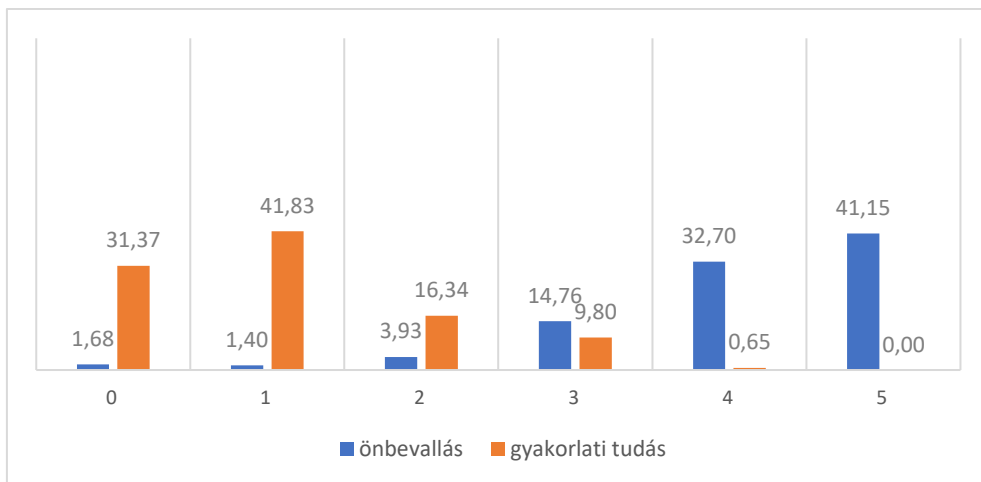
Az összehasonlítás során a hibakeresés és felismerés folyamatát értékeltük, a javítást és a helyes formázást nem.

	évfolyam	
	7.	9.
szintaktika	40,72%	42,92%
tipográfia	0,00%	0,00%
tördelés	6,11%	23,14%
összesen	20,29%	28,62%

6. táblázat Az ERM mérés előtesztjének eredményei évfolyam és hibatípus szerinti bontásban.

Az eredmények a 6. táblázatban is látható 3 hibatípusba sorolhatók, így ezek különálló eredményei is beszédesek. A tanulók évfolyamtól függetlenül a szintaktikai hibákat ismerték fel legnagyobbbrészt (40,72% és 42,92%), amelyek felismerése elsősorban magyar nyelvi tudást igényel és nem informatikai jártasságot. A másik két hibatípusban mindkét évfolyam alacsonyabb eredményt ért el, azonban a tördelési hibákat szignifikáns mértékben magasabb százalékban ismerték fel a magasabb, 9. évfolyamos tanulók ( $p=0,0000$ ).

Az ERM mérés előtesztjén kapott eredményeket a mini kompetencia tesztel való összehasonlítás céljából átalakítottuk és megfeleltettük egy 0-5-ig terjedő skálának (24. ábra). Az önértékelés (mini kompetencia teszt) és a gyakorlati tudás (ERM előteszt) mértéke nincs összhangban, sőt fordítottan arányos. A felmérések eredményei közötti különbség mintapéldája a Dunning-Kruger-hatásnak (Kruger & Dunning 1999), továbbá igazolja a [T4] tézist. Ez az el-lentmondás rámutat arra is, hogy a tanulók az egyes jártassági terület alatt, mint a szövegkezelés vagy táblázatkezelés, elsősorban a programkörnyezetre és az azon történő navigációra gondolnak. A gyakorlatban ezzel szemben nem a környezetben való navigálás, eligazodás számít, hanem, hogy a felhasználó képes-e hibáktól mentes, jól-szerkesztett dokumentumot tervezni és létrehozni.



24. ábra A mini kompetencia önbevalláson alapuló és az ERM gyakorlati tudást mérő vizsgálat eredményeinek összehasonlítása.

### 6.3. Táblázatkezelés

A munkavállalók felé elvárás, hogy készség szinten használják a szöveg- és táblázatkezelő alkalmazásokat. A legtöbb felhasználó azonban nem képes hibamentes dokumentumot készíteni, szerkeszteni. A tudásuk elsősorban a navigációban, a programkörnyezet ismeretében merül ki. Több esetben ez nem számít, a munkáltatónak is elegendő, ha a készített dokumentum nyomtatásban jól néz ki. Azonban vannak területek, mint az üzleti és a gazdasági szféra, ahol a táblázatkezelő programok helyes gyakorlati alkalmazása nélkülözhetetlen. Az esetleges hibák ezekben az ágazatokban komoly károkat, elsősorban anyagiakat okozhatnak. A Ray Panko mérései alapján a táblázatkezelői dokumentumok legalább 86%-a hibás (Panko 2008), tekintve a világ Excel és más táblázatkezelők használóinak és az általuk létrehozott dokumentumoknak a számát, igen nagy mennyiségről van szó. Azonban ezeknek a hibáknak a megléte nem csak a kisvállalatok mérlegeinek torzulásához vezet, hanem globális szinten is érezhető a hatása (Abraham & Erwig 2009, Jorgensen 2013, Kadijevich 2009, 2013, Kwak 2013, Panko & Aurigemma 2010, Panko 2008, Powell et al. 2009, 2009a, 2009b, Teo & Tan 1999, Tort 2010, Tort et al. 2008, W2

2012). Az EuSpriG (European Spreadsheet Risks Interest Group) horror sztorik néven gyűjt különböző táblázatkezelői hibás dokumentumból, képletből adódó eseteket, amelyek több millió dollár kárt okoztak már a világon (EuSpriG 2020). A hibákból eredő károk nem feltétlenül közvetlenül anyagi vonzataiak, egy-egy elrontott táblázat és számítás hatással lehet egész ország gazdasági stratégiájára. Két harvardi professzor kapcsolatot mutatott ki az országok adóssága és gazdasági növekedése között, ennek hatására több ország megszorításokat vezetett be, amely több millió ember életére volt hatással. Több közgazdász próbálta reprodukálni az eredményeket sikertelenül, csak három év eltelte után tették az eredeti táblázatokat nyilvánossá, ahol kiderült, hogy számos számítási hiba található benne (Krugman 2013).

A táblázatkezelés problémái megegyeznek más alkalmazói ismeretek nehézségeivel. A felhasználók elsősorban próbálgatással, másnéven TAEW (Trial-and-error-wizard-base) típusú megközelítéseket alkalmaznak, melynek a legalacsonyabb (1) a mathability szintje (Baranyi & Gilányi 2013, Borus & Gilányi 2013). Ezt az attitűdöt támogatják a szoftvergyártó cégek is, akik „felhasználóbarát” szlogen mögé rejtve biztatják a felhasználókat az ész nélküli kattintgatásra. A legnagyobb probléma, hogy ezeknek a szoftvereknek a használata során a használók előzetes tervezés nélkül dolgoznak, valamint, hogy a kapott eredményt sem ellenőrzik. Örülnek annak, ha bármilyen eredmény kijön, függetlenül attól, hogy az helyes-e (Ben-Ari 1999, Csernoch 2009, 2010, Csernoch & Biró 2015a, 2015b, 2018, Sebestyén et al. 2021).

A szövegkezeléssel ellentétben a táblázatkezelő program használata összetettebb, hisz itt nem csak a külalak számít, hanem a pontos számítások is. A függvények alkalmazásakor elengedhetetlen a függvényalkotás lépéseinek és helyes szintaktikájának ismerete. A táblázatkezelés oktatását a Kerettanterv (OFI 2012b, 2020) és a tankönyvek (Bártfai 2012, Dancsó & Korom 2013, Farkas 2009, Lakosné 2019c, Varga et al. 2010) tovább nehezítik, a bennük

lévő mintafeladatok sok esetben csak 4-5 rekordból álló, tartalom és összefüggés nélküli táblázatok, ahol a gépelés is a feladat része. A segédanyagok több mint 100 függvényt vonultatnak fel (Bártfai 2012), amelyeknek az elsajátítása a megadott óraszám mellett és a felületalapú megközelítéssel lehetetlen, ráadásul felesleges is. A szoftverekben ugyan megtalálható a függvények leírása és alkalmazásának feltételei, azonban ennek az értelmezéséhez szükséges matematikai, gazdasági ismeretekkel nem csak a diákok, de a pedagógusok többsége sem rendelkezik.

### **6.3.1. Spreadsheet Lego (Sprego)**

A Sprego módszer létrejött a Nemes Tihamér versenyekhez köthető, hiszen a táblázatkezelői feladatmegoldás során „programozási ismereteket” kértek számon, amit a hagyományos, felületalapú módszerekkel a tanulók nem tudtak megoldani. A módszer alapjait Csernoch Mária tette le, amit az évek oktatási tapasztalatai tettek teljessé (Csapó & Sebestyén 2015, 2017; Csapó et al. 2020, 2021; Sebestyén et al. 2019).

A módszer a Spreadsheet Lego szavak összevonásából áll, amely arra utal, hogy a táblázatkezelői függvényeket lego módjára építi össze. A megközelítés Pólya György (1954) 4-lépéses problémamegoldási stratégiáján alapul, valamint a programozásoktatásban is alkalmazott és bevált séma- és algoritmus-építésen. A módszer különlegessége, hogy a tankönyvi példákkal és a gyártócégek ajánlásával ellentétben néhány általános célú függvényt alkalmaz a több száz összetett függvénnyel szemben, amelyek egymásba ágyazva – mint a matryoska baba – komplex problémák megoldására is alkalmasak. Az általános- és középiskola kimeneti követelményeit a 12 általános célú táblázatkezelői függvény lefedi (7. táblázat), azonban igény esetén tovább bővíthető az alkalmazott általános célú függvények listája (Csernoch & Biró 2015b).

Sprego szöveg	Sprego szám	Sprego pro
BAL()	MIN()	HA()
JOB()	MAX()	INDEX()
HOSSZ()	SZUM()	HOL.VAN()
SZÖVEG.KERES()	ÁTLAG()	HIBÁS()

7. táblázat A Sprego módszer által használt általános célú függvények.

A nemzeti alaptantervvel, kerettantervvel (OFI 2012a, 2012b, 2020) és a tankönyvekkel ellentétben a Spregoiban fontos szerepet kapnak a szövegfüggvények. Az elmúlt évek tapasztalata alapján elmondható, hogy a módszer már egészen fiatal korban bevezethető és alkalmazható, mégpedig a szövegfüggvényeknek köszönhetően. Ennek oka abban keresendő, hogy a tanulók nehezen boldogulnak a matematikai feladatokkal, számításokkal, alacsony az algoritmizálási képességük – hogyan is lenne magasabb, ha erre az oktatási rendszer nem készíti fel őket? –, így már a pusztán gondolat, hogy számokkal kell dolgozni negatívan hat a tanulók motivációs szintjére.

A felhasznált források mindegyike a valóéletből vett, autentikus, nem csak 4-5 rekordból álló táblázat. A táblázatok választásakor figyelembe vesszük a tanulók életkori sajátosságait és érdeklődési körét. Közkezdelt témák, táblázatok közé tartozik az imdb top 250-es film listája (IMDb 2019), a youtuberek rangsora, a League of Legends<sup>10</sup> ranglistája, amelyekben a tanulók szívesen keressélnék, nézelődnek és örülnek, ha találnak egy-egy ismerős címet, nevet.

A módszer támogatja a sémaépítést és az intelligens tanulást (Merriënboer & Sweller 2005; Skemp 1971), ezek mindegyike elősegíti a hosszútávú tudás elérését (Carr 2011; Kahneman 2011; Csernoch & Biró 2015a; National Research Council 2000), amely az analitikus, lassú és a gyors gondolkodáson alapszik (Csernoch 2017; Kahneman 2011). A tanulási folyamat során a tanulók egy tucat függvényt használnak és alkalmaznak különböző módokon, az egy-

---

<sup>10</sup> számítógépes játék

szerű problémáktól a komplexekig. Az alkalmazott függvények többszöri alkalmazása, új kontextusba való helyezése egyfajta állandóságot, biztonságot nyújt a tanulóknak, ezáltal hatékonyabban tudnak haladni (Kátai et al. 2016; Osztián et al. 2017).

### Feladatok

1. A bal oldali ábra az Irka Iskola 9. b osztályának éves pénzforgalmát tartalmazza. Az osztálypénz szeptember 1-jén 0 Ft-tal indult, utána havi bontásban látjuk a bevételek és kiadások összegét. Készítsük el a táblázatot! A hónapok oszlopát kitöltéssel vigyük be. Milyen képlet került a E4-es cellába?

	A	B	C	D	E
	<b>Osztálypénz - havi bontásban</b>				
1					
2			<b>Bevétel</b>	<b>Kiadás</b>	<b>Egyenleg</b>
3		Aug			0
4		Szept	80000	7800	72200
5		Okt	120000	13700	178500
6		Nov	40000	23780	194720
7		Dec	20000	13300	201420
8		Jan	10000	1800	209620

	A	B	C	D	E
1	Benzinár:	380	Fogyasztás:	7,4	l/100 km
2					
3	<b>mikor</b>	<b>honnan</b>	<b>hová</b>	<b>út (km)</b>	<b>költség (Ft)</b>
4	hétfő	Budapest	Kaposvár	190	5342,8
5	kedd	Kaposvár	Répcelak	179	5033,48
6	szerda	Répcelak	Tófej	102	2868,24
7	csütörtök	Lenti	Siófok	164	4611,68
8	péntek	Siófok	Tyukod	427	12007,24
9		<b>Összesen:</b>			<b>29863,44</b>

25. ábra Digitális kultúra tankönyv 9 egyik táblázatkezelői feladata, ahol több a gépelés, mint a valós feladat (Varga et al. 2020).

Nyissa meg a táblázatot!

A táblázat gyógyszerek értékesítési adatait tartalmazza az A2:J35 tartományban a következő adatokkal: termék neve, kiszerelés, termékkód, értékesített mennyiség 2007–2013-ig.

	A	B	C	D	E	F	G	H	I	J
1										
2	Termék	Kiszerelés	Termékkód	2007	2008	2009	2010	2011	2012	2013
3	ROLICADE 4	12*1 L	YCB064	5851	3125	2000	2600	2000	1500	1000
4	ROLICADE 4	12*1 L	YCB110	17918	13200	16000	15500	14000	13000	12000
5	ROLICADE 4	12*1 L	YCB276	13979	6443	12000	13000	13000	13000	13000
6	ROLICADE 4	4*5 L	YCC132	12820	11200	15000	16000	16000	16000	16000
7	KORDEXIN	4*5L	YCT315	0	3160	15000	18000	20000	21000	21000
8	KORDEXIN	12*1L	YCK410	0	1920	3000	5000	6000	7000	7000
9	KORDEXIN	4*5 L	YCK411	13635	10265	0	0	0	0	0
10	KORDEXIN	4*5 L	YCB483	13635	15345	18000	23000	26000	28000	28000
11	KORDEXIN	25 L	YCK448	3975	1350	0	0	0	0	0
12	KORDEXIN	40*125 L	YCC020	1652	1530	1500	1400	1300	1200	1100
13	HARSO	4*5 L	YCS546	6440	7195	12500	11500	10700	10300	9900

Feladata a következő:

1. Szűrjön be egy új terméket a jelenleg első helyen szereplő termék elé, lássa el tetszőleges adatokkal! [2 pont]
2. Készítsen a lista végére (a 37. sorba) egy összegző sort, amelyben az évenként értékesített összmennyiséget számolja ki! [2 pont]
3. Készítsen el egy átlagot mutató sort az összegző sor alá (a 38. sorba)! [2 pont]
4. A K oszlopban számítsa ki függvényel minden termékre a 2010 utáni évekhez tartozó értékesítési adatok összegét! [2 pont]

26. ábra ECDL vizsgapéldatár táblázatkezelés 1. feladatlapjából részlet, ahol a megoldás minden lépése pontról pontra meg van adva. A tanuló részéről nem igényel számítógépes gondolkodást, csupán a lépések követését. (ECDL Foundation)

A táblázatkezelés oktatása a Sprego módszerrel az évek alatt sokat finomodott, azonban már az első perctől kezdődően hatékonyan alkalmazható volt.

A tanórákon a tankönyvekből ismert feladatok, amilyen a „minta alapján való-sítsd meg” (25. ábra) vagy az ECDL és érettségi feladatokra is jellemző „szakácskönyv” jellegű (26. ábra) feladatok helyett, a problémamegoldás és séma-építés került középpontba.

A Sprego módszer alkalmazására, a szövegfüggvények bevezetésekor alkalmazott problémafelvetésre mutatok egy példát. A témakör bevezetésekor kiemelten fontos a tanulók táblázattal kapcsolatos ismereteinek előhívása, va-lamint bővítése. Új fogalmakkal és formátumokkal ismerkednek meg a tanulók a táblázat elemzése során, minden oszlop vizsgálatára sor kerül, ahol megné-zik, hogy milyen adatok szerepelnek és hogy azok milyen típusúak (27. ábra). A C „Title” oszlopban rögtön feltűnik a tanulónak, hogy szöveget és számot is tartalmaz, ami nem csak két típus, hanem két különböző adatot is rejt, a címet és az évszámot. Ennek köszönhetően adott a probléma, hogy külön kell válasz-tani a címet és az évszámot.

	A	B	C	D
1	Rank	Rating	Title	Votes
2	1	9,2	A remény rabjai (1994)	727260
3	2	9,2	A keresztapa (1972)	544185
4	3	9	A keresztapa 2. (1974)	342859
5	4	8,9	Ponyvaregény (1994)	571887
6	5	8,9	A jó, a rossz és a csúf (1966)	227552
7	6	8,9	Tizenkét dühös ember (1957)	178266
8	7	8,9	Schindler listája (1993)	382118

27. ábra Az IMDb (Internet Moovie Database) top 250 filmjét tartalmazó listája, ami az internetről származó fájlkonverzióval (WDC módszer) létrehozott táblázat jelenít meg (IMDb 2019).

Kezdeként az évszámot szeretnénk kiírni, így erre irányítjuk a tanulók fókuszát. Néhány egyszerű kérdéssel hívjuk fel a tanulók figyelmét az azonos-ságokra. A tanulók így felfedezik, hogy minden évszám a cím után, a szöveg jobb oldalán helyezkedik el, valamint azonos hosszúságúak. Az elemzés során minden oszlop esetében meghatározásra került az adatok típusa – szöveg és szám –, így az évszám esetében is, azonban a zárójelek miatt a táblázatkezelő

szövegnek érzékeli. Ez generál egy újabb problémát, amit meg kell oldaniuk a tanulóknak. Ahogy a természetes nyelven érkező megoldási javaslatokat, úgy később a táblázatkezelői képleteket is lépésről lépésre pontosítjuk és bővítjük (8. táblázat).

lépés	képlet	kiíratás eredménye
1.	<code>{=jobb(C2:C251;5)}</code>	évszám és zárójel: XXXX)
2.	<code>{=bal(jobb(C2:C251;5);4)}</code>	évszám, de még szöveg formátumban
3.	<code>{=bal(jobb(C2:C251;5);4)*1}</code>	évszám, szám formátumban

8. táblázat Az évszám kiíratásának lépései Sprego módszerrel.

#### 6.4. Oktatási segédeszközök

A tanítási tapasztalataink alapján, a függvények egymásba ágyazása, az algoritmusépítés lépéseinek elsajátítása gyorsabban megy végbe, ha az valamilyen unplugged, semi-unplugged eszközzel támogatott. A táblázatkezelés oktatása során az első perctől kezdve megjelennek az unplugged elemek, mint a matrjoska baba, origami hajók, 3D nyomtatott baba szett és hordók, valamint ezek táblai megfelelői. A gyakorlatban a legkönnyebben beilleszthető és kivitelezhető eszköz, az origami hajók használata, hiszen néhány papírlapra van szükség a hajók meghajtásához – ami kíváncsivá, érdeklődővé teszi a tanulókat –, valamint beragasztható a füzetbe, így később is megnézhető, tanulmányozható.

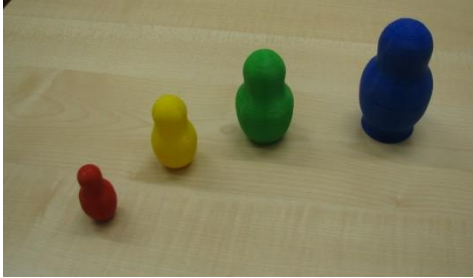
A témakör elején szövegfüggvényekkel ismerkednek meg a tanulók, mint a 6.3.1 fejezetben ismertetett évszám kiíratása, amely több lépésből áll, a képlet folyamatosan bővül (8. táblázat). A tanulók meghajtják a különböző méretű origami hajókat (28. ábra). A problémamegoldás során mindig van egy bemenő adat. Az évszám leválasztás példáján szemléltetve, ez az évszám és a cím (C oszlop), ezt a meghajtott hajó belsejébe írják a tanulók. A hajó egyik külső oldalára kerül az algoritmus lépése – természetes nyelven vagy képlettel, előrehaladástól függően –, míg a másik oldalra a várható kimenet. Kiemelten

fontos az eredmény ellenőrzése és debuggolása, és ebben már az előre átgondolt és várt kimenet megfogalmazása és rögzítése segítséget nyújt (8. táblázat). A következő, eggyel nagyobb méretű hajó egyben az algoritmus következő lépését jelenti. A következő lépés bemenő adata nem lesz más, mint az első hajó kimenete, így csak beletesszük a kishajót a nagyobbba. Ezt követően megfogalmazzák a tanulók a következő lépést és a várható kimenetet, amit a hajó külső oldalán rögzítenek. Annyi hajóra van szükség, ahány lépésből áll az algoritmus, az évszám leválasztása során ez 3. A legtöbb probléma 2-3 lépésből áll, amiket a tanulók eltehetnek a füzetükbe.



28. ábra Unplugged oktatási segédeszköz a Sprego módszerhez. A 4 különböző méretű origami hajó szemlélteti az algoritmus 4 lépését, amelyeket a legkisebbtől haladva építünk és bővítünk a hajókat egymásba téve.

Az algoritmusépítésre egy másik unplugged variáció a 3D nyomtatott baba szett. Minden tanuló kap egy 3-4 babából álló készlet (29. ábra). Ennél az eszközknél a tanulók ragasztószalagot használnak az algoritmus lépéseinek, a bemenő és kimenő adatok leírására. A legkisebb babába kerül a bemenet, majd az algoritmus lépését, és a kimenő adatot a ragasztószalagra írva ráragasztjuk a babára és így tovább (31. ábra). A probléma megoldása után a ragasztószalagok beragasztásra kerülnek a füzetbe (30. ábra).



29. ábra 4 darabból álló 3D nyomtatással készült baba szett.



31. ábra A 3D babák használat közben, ragasztalagon az algoritmus lépése.



30. ábra Füzetkép a problémamegoldás során használt ragasztószalagok beragasztása után.

### 6.4.1. 2D oktatóprogram

Csapó Gáborral úgy gondoltuk, hogy a képletek egymásba ágyazása és lépései néhány tanulónak gondot okoz (alacsony algoritmizálási képesség), így az ő megértési folyamatukat szerettük volna elsősorban segíteni. Az oktatóprogram ötlete, célja a megvalósítás közben folyamatosan alakult és kiegészült. A program már nem csak a tanórai, hanem az otthoni tanulási folyamatba is beépíthetővé vált (Csapó & Sebestyén, 2015, 2017, Sebestyén & Csapó 2018, Sebestyén et al. 2019).

Az oktatóprogram tervezése során a feladatokat megosztottuk egymás között, az én feladatomban a felület tervezése, a program kidolgozása és grafikai megvalósítása volt, míg Csapó Gábor a program életre keltését, fejlesztését valósította meg a Construct 2 program segítségével. A tervezés és a kivitelezés

folyamatát témavezetőnk Csernoch Mária felügyelte. A program alkalmazása során kapott tanulói és tanári visszajelzéseket tanulmányoztuk, a tanácsokat megfogadtuk és integráltuk a hatékony alkalmazás érdekében.



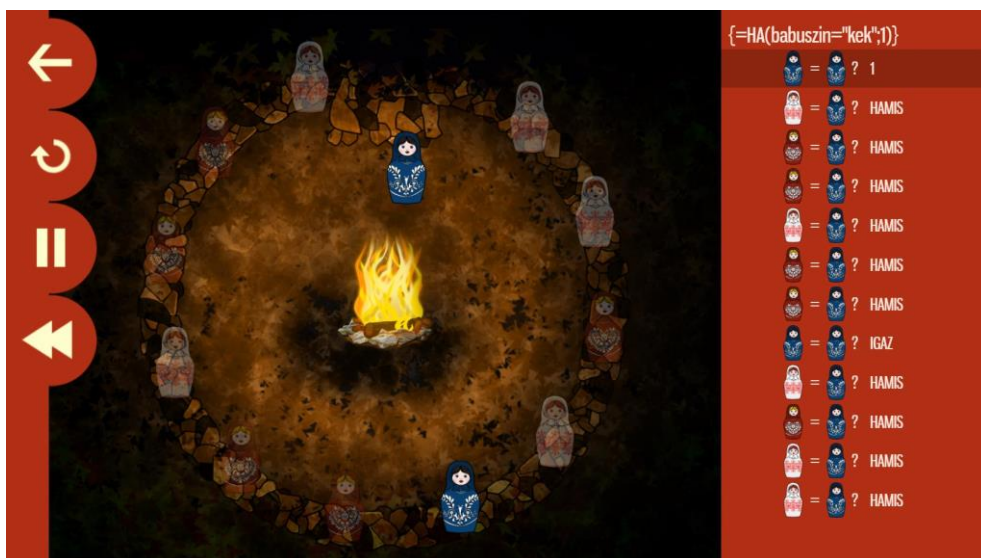
32. ábra Sprego oktatóprogram kezdőképernyője, ahol kiválasztható a lejátszani kívánt probléma, valamint jobbra a problémához tartozó beállítási lehetőségek.

A program felépítése két részből áll, egy kezdőképernyőből, ahol kiválasztható a probléma és megadhatók az animáció paraméterei (32. ábra), valamint egy animációs felületből, ahol a játékos megjelenítés mellett, párhuzamosan egy képletkiértékelő is szemlélteti az algoritmus lépéseit. A kezdőképernyőn és az animációs felületen is találhatóak akciógombok, a kezdőképernyőn lehetőség van hang, zene be- és kikapcsolására, valamint nyelvválasztásra – magyar és angol nyelven érhető el.

Az animációs felület nagyobb részét a képi animáció foglalja el, melynek minden esetben matrjoska babák a szereplői, amely a Sprego módszer jelképe, emellett egy képletkiértékelőt is elhelyeztünk, ami párhuzamos mutatja az algoritmus lépéseit a szemléltető animációval együtt (33. ábra).

A Google Play-en is elérhető Sprego oktatóprogramban 3 táblázatkezelői probléma kerül bemutatásra, a feltételes összegzés, a lineáris keresés, valamint

egy egyenlőtlenségi probléma. Mindezekon túl további problémák animációjának algoritmusai és grafikája került megtervezésre, azonban a programba egyelőre nem került implementálásra.



33. ábra A Sprego oktatóprogram animációs ablaka, „Hány azonos színű baba van?” probléma, kék babaszín paraméter választásával.

A problémák kiválasztása tudatos, a feltételes összegzés a Sprego módszer szerint a =SZUM(HA()) függvények egymásba ágyazásából áll (33. ábra), míg erre a tankönyvek és szoftvergyártók a problémáspecifikus =SZUMHA(), =DARABTELI(), =DARAB() – és még sorolhatnánk – függvényeket ajánlják. Minden problémára van egy függvény, ami másra nem használható, csak annak az egy problémának a megoldására, így nem csoda, hogy több mint 500 függvény található a táblázatkezelői rendszerekben. Ez a függvény mennyiség nem csak a tanulók munkamemóriáját haladja meg, hanem a magasabb algoritmizálási képességgel rendelkező felhasználóké is.

A lineáris keresés a Sprego módszer szerint =INDEX(HOL.VAN()) függvénypárossal történik (34. ábra), amely helyett a közoktatásban =FKERES() és a VKERES() függvények használata terjedt el. Az utóbbi függvények használata korlátozott, fontos a rendezés, valamint a keresési és a találati vektorok

egymáshoz viszonyított helyzete és a keresés iránya. Ezzel szemben a Sprego megoldása univerzális és kötetlenebb.

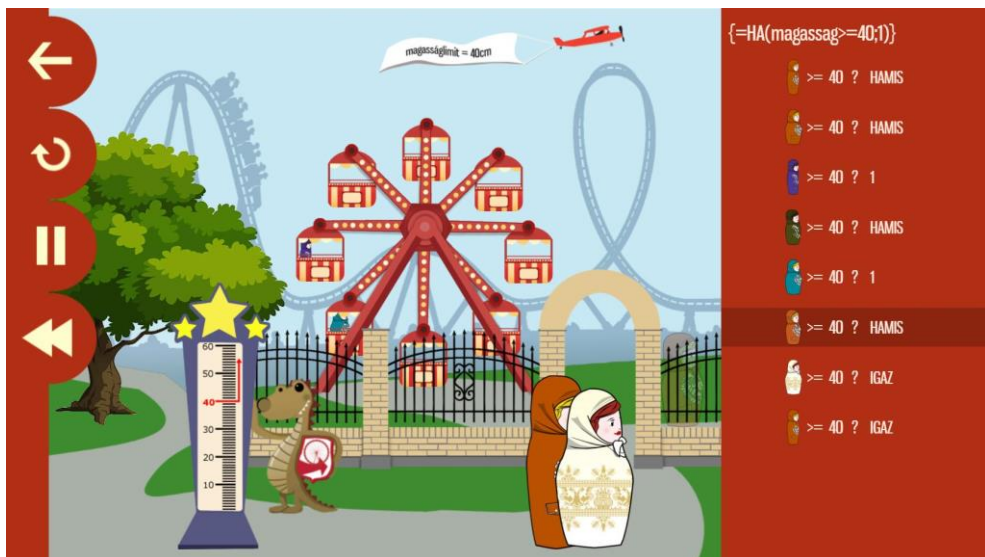


34. ábra A Sprego oktatóprogram animációs ablaka, „Melyik házban lakik a baba?” probléma, arany babaszín paraméter választásával.

A harmadik beépített probléma szintén egy feltételes összegzés, azonban egyenlőtlenségi esetben (Sebestyén & Csapó 2018). A Sprego =SZUM(HA() függvények esetében nincs különbség egyenlőség és egyenlőtlenség esetében (35. ábra), azonban a közoktatásban elterjedt problémáspecifikus függvények esetében a szintaxis különbözik egymástól, amit a =DARABTELI() függvény segítségével mutatok be a Microsoft hivatalos példái alapján (Microsoft 2020).

- =DARABTELI(A2:A5;10); egyenlőség konstanssal
- =DARABTELI(A2:A5;A4); egyenlőség változóval
- =DARABTELI(A2:A5;">85"); egyenlőtlenség konstanssal
- =DARABTELI(A2:A5;"<"&D1); egyenlőtlenség változóval

A tanulóknak tehát nem elég az argumentumok listáját és sorrendjét megjegyezniük, hanem arra is figyelniük kell, hogy egyenlőségi vagy egyenlőtlenségi esetet oldanak meg, és hogy azt konstanssal vagy változóval teszik.

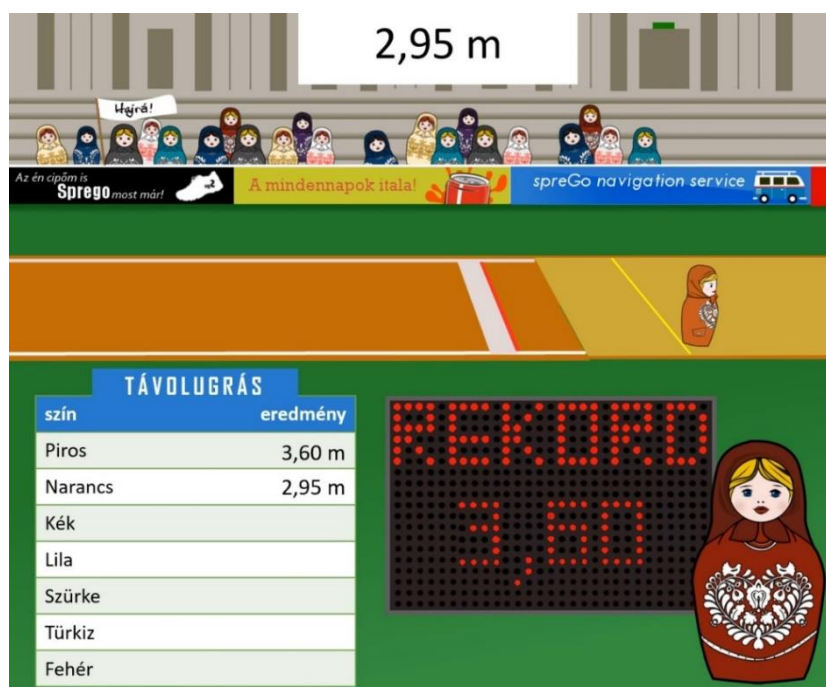


35. ábra A Sprego oktatóprogram animációs ablaka, „Hány baba ülhet fel az óriáskerékre?” probléma, 40 magasságlimit és  $\geq$  paraméter választásával.

Az elterjedt problémaspecifikus függvények használatával, mint a SZUMHA, DARABHA, ÁTLAGHA a feltétel megadása egyenlőtlenség esetén sztringként működik – idézőjelek között –, amely a diákok adattípusra vonatkozó ismereteit összezavarhatják. Ezzel szemben a Sprego által támogatott általános célú függvények használata következetes és nem tesz különbséget egyenlőség és egyenlőtlenség között. A komplex problémák megoldása mellett igény mutatkozott a 12 Sprego függvény önálló bemutatására, így ezek közül kidolgozásra került a SZUM(), ÁTLAG(), HOL.VAN(), INDEX(), MIN() és MAX() függvény (36. ábra). A függvények ábrázolásához használt problémák, környezetek kiválasztására kiemelten figyeltünk.

A tervezési folyamat alatt a függvények algoritmusának lépéseire nagy hangsúlyt fektettünk, hiszen ez a program alappillére. Fontosnak tartottuk, hogy a programot használók egy könnyen érthető, mindennapi életből vett példákon ismerjék meg a függvényeket. A függvények külön-külön történő ábrázolása, bemutatása lehetővé teszi a program használatát már a kezdetektől. A

problémák tervezése, grafikai megvalósítása megtörtént, az implementálásra egyelőre nem került sor.



36. ábra A Sprego oktatóprogram animációs ablakának terve a „Melyik baba ugrik a legnagyobb?” probléma szemléltetéséhez, amely a MAX függvény működését szemlélteti.

A Sprego oktatóprogramban megjelenő problémák esetén törekszünk az egyediségre. A tanulási folyamat hatékonyságának növelését, az ismeretek rögzítését segítik az eltérő környezetű problémák (utca, vidámpark, sportpálya), szituációk alkalmazása. A különböző környezeteknek köszönhetően a tanuló képes lesz egy-egy témához kötni a függvényeket, ezáltal könnyebben visszaidézi a problémák algoritmusát is.

## 6.5. Szerepvállalás a kutatócsoportban

A kutatócsoport tagjaként a TAaAS felmérés értékelésében, javításában és az eredmények digitalizálásában és publikálásában vettem részt.

Aktívan szerepet játszottam a Sprego, a Fájlkonverzió és a Hibafelismerés módszerek hatékonyságának növelésében, segédeszközök fejlesztésében

(Sprego 2D oktatóprogram) és gyakorlati alkalmazásában, tesztelésében, a módszerek népszerűsítésében (Informatikai Szakmai Napok, Digitális Témahét, Science on Stage). 2016-tól kezdődően az egyetemi tanulmányaim mellett teljes, majd később részmunkaidőben informatikatanárként dolgoztam egy gimnáziumban, ahol a Sprego mellett a korábban említett Fájlkonverzió és Hibafelismerés módszerekkel is tanítottam a tanulókat. A módszerek mindegyikéhez hatékonyságvizsgálat társult, így ezeknek az előkészítésében, feladatlapjainak összeállításában, majd a tanulócsoportok tesztelésében, oktatásában, valamint az eredmények javításában, digitalizálásában és statisztikai elemzésében vettem részt.

## **7. Hatékonyságvizsgálat és eredményei**

A [T1] tézist képező 3 informatikai témakör mindegyikéhez kidolgoztuk a módszerek hatékonyságvizsgálatának mérését, melyhez a vizsgálati csoport oktatójaként a kutatócsoport tagjai és a kontroll csoport vezetőjeként a kutatócsoporttól független informatikatanárok csatlakoztak. Az iskolai informatika csoportok létszáma alacsony a különböző csoportbontások miatt, így átlagosan egy csoport 10-18 fő közé tehető. Az alacsony tanulói létszám és a közreműködő pedagógusok száma miatt a hatékonyságvizsgálati méréseken a minta nagysága ~100 fő körül mozog.

### **7.1. Fájlkonverzió (WDC)**

A módszer hatékonyságvizsgálatát 2018/2019-es tanévben végeztük el két középiskolában (Sebestyén 2020). A vizsgálatban résztvevő valamennyi tanuló 9. évfolyamos, mind a kontroll-, mind a vizsgálati csoport esetében. A fájlkezelés témakörét a korábbi évfolyamokon 6. évfolyamtól kezdődően a

Kerettanterv szerint több alkalommal tanulták. A módszer hatékonyságvizsgálata során minden tanuló a tanítási folyamat előtt és után tesztet írt (1. számú melléklet), azonban néhány tanuló esetében iskolai elfoglaltság, betegség miatt nem volt lehetséges mindkét teszt megoldása, így az ő fejlődésük nem számszerűsíthető. További tanulói tesztek váltak használhatatlanná azáltal, hogy két csoport esetében a kontroll csoportot tanító tanár kilépett a vizsgálatból, így a 9. táblázatban látható módon alakult a vizsgálatban résztvevő tanulók létszáma.

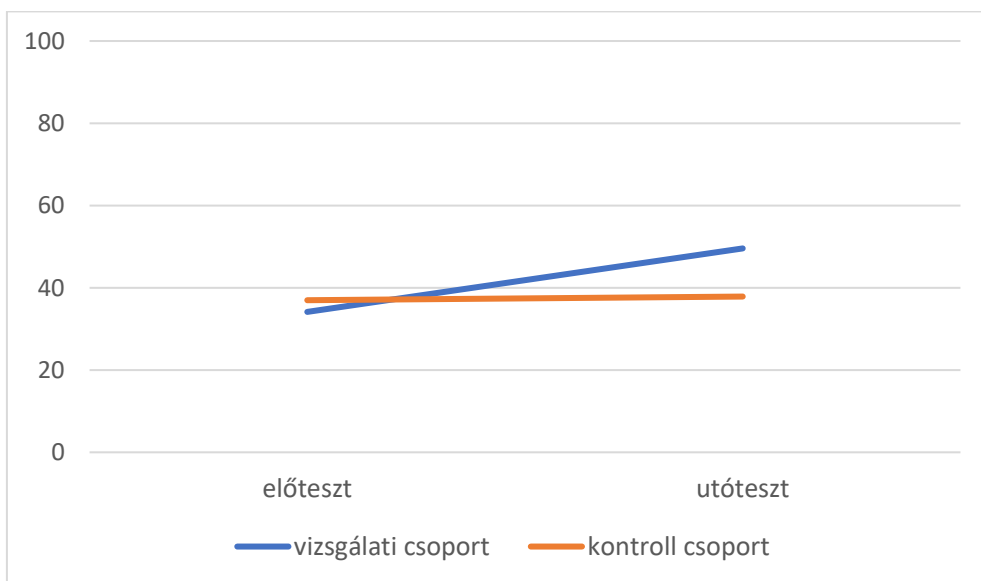
	<b>vizsgálati csoport</b>	<b>kontroll csoport</b>
előteszt	30	79
utóteszt	35	51
párosított teszt	28	45

9. táblázat WDC módszer hatékonyságvizsgálatán résztvevő tanulók száma és eloszlása.

Az előteszt során a csoportok között nincs szignifikáns eltérés ( $p=0,0607$ ). A vizsgálati csoport az előteszten 33,73%-ot ért el, míg a kontroll csoport 38,02%-ot. Az előteszt eredményei az előző 6.1 fejezetben feladatonként olvasható, azonban ott csoportbontás nem szerepelt, mert egy általános képet szeretnénk volna kapni a középiskolát megkezdő 9. évfolyamos tanulók témában hozott ismereteiről.

Az oktatási folyamat végén a tanulók az előteszt feladatlapját töltötték ki ismételten, amellyel az oktatási-tanulási folyamat alatt szerzett és rendszerezett ismereteikről adtak számot. A tanulói eredmények változását, fejlődését minden esetben a tanulók saját előtesztjével hasonlítottuk össze, tehát a csoportok fejlődését önmagukhoz mértük, így csak azoknak a tanulóknak az eredményeit vettük figyelembe a végső összehasonlításnál, akik mind elő-, mind utótesztet írtak (9. táblázat, párosított teszt). A 37. ábra kiválóan szemlélteti, hogy a vizsgálati csoport az utóteszten szignifikánsan magasabb eredményt ért el, mint az előteszten ( $p=0,0000$ ). A hatékonyságvizsgálat statisztikai elemzése

során az igazán meglepő, nem az, hogy a vizsgálati csoport az utóteszten szignifikánsan jobban teljesített, mind önmagához, mind a kontroll csoport eredményéhez mérten, hanem, hogy a kontroll csoport, az előteszt 38,02%-hoz képest az utóteszten ennél alacsonyabb, 37,88%-osan teljesítette a tesztet ( $p=0,5795$ ). Ez az eredmény nem csak a WDC módszer hatékonyságát igazolja, hanem megkérdőjelezi a hagyományos, felületi navigációra épülő módszerek, feladatok hatékonyságát. Továbbá felveti azt a kérdést, hogy mi történt a tanulási folyamat alatt, milyen ismeretekkel lettek gazdagabbak a kontroll csoport tagjai. A 37. ábra és a csoportok összeredményei egyértelműen igazolják a [T1] tézist, mely szerint az algoritmusépítésre épülő módszerek, amilyen a WDC is, szignifikánsan hatékonyabbak, mint a hagyományos, low-mathability megközelítések.



37. ábra WDC hatékonyságvizsgálat elő- és utótesztjén elért eredmények a vizsgálati és a kontroll csoportban.

## 7.2. Hibafelismerési modell (ERM)

A Hibafelismerési modell hatékonyságának számszerűsítése és bizonyítása érdekében vizsgálati csoportokat teszteltünk, ahol az újszerű, high-mathability (Baranyi & Gilányi 2013; Biró & Csernoch 2015, Sebestyén et al. 2021) ERM módszerrel ismerkedtek meg a tanulók a szövegkezeléssel. Az eredményeket olyan kontroll csoportokkal hasonlítottuk össze, ahol a hagyományos kerettantervre épülő, felület- és eszközalapú módszereket alkalmazták a pedagógusok. Az oktatási folyamat és a tesztelés a 2017/2018-as tanévben zajlott egy debreceni középiskolában. A vizsgálat résztvevői 7., valamint 9. évfolyamos diákok. A Kerettanterv szerint ezek a tanulók a korábbi évfolyamokon már elsajátították a szövegszerkesztés alapjait, képesek szöveges dokumentumokat elkészíteni minta alapján. Vizsgálatunk során a korábbi hatékonyságvizsgálatokhoz hasonlóan két körben teszteltük a tanulókat, az oktatási folyamat elején és a végén. A hatékonyságvizsgálatban összesen 153 fő vett részt, azonban a tanulói hiányzások, iskolai elfoglaltságok miatt összesen 130 fő volt az, aki mindkét tesztelésen részt vett (10. táblázat) (Sebestyén et al 2021).

	előteszt		utóteszt		párosított teszt	
	vizsgálati csoport	kontroll csoport	vizsgálati csoport	kontroll csoport	vizsgálati csoport	kontroll csoport
7. évf.	34	-	29	-	26	-
9. évf.	69	50	73	44	66	38
fő/csoport	103	50	102	44	92	38
összesen	153		146		130	

10. táblázat A résztvevő tanulók alakulása az ERM hatékonyságvizsgálat elő- és az utótesztjén, ezen belül évfolyam- és csoportbontásban.

A tesztelés a korábban ismertetett 3 szakaszból állt (6.2.1), erre a tanulóknak 45 perc állt rendelkezésükre. Az előteszt és az utóteszt során alkalmazott dokumentum egymástól eltérő (2. számú, 3. számú melléklet), azonban a bennük található hibák és azok típusa megegyezik (11. táblázat).

hibatípus	hiba	észlelés helye
szintaktikai hiba	helyesírási hibák (ET/UT) zárójelek és szóközők használata (ET/UT) hiányzó szóköző (UT)	nyomtatott dokumentum
tipográfiai hiba	aláhúzás (ET/UT) dőlt betűstílus (ET/UT) félkövér betűstílus (UT) nagybetűs betűstílus (UT)	nyomtatott dokumentum
tördelési hiba	üres bekezdések (ET/UT) bekezdés vége jel a sorok végén (ET/UT) behúzás szóközőkkel (ET/UT) igazítás szóközőkkel (ET/UT) kézi elválasztás (ET/UT) kézi számozás (UT) több szóköző (UT) ritkítás szóközővel (UT)	digitális dokumentum

11. táblázat Az elő- és az utóteszt szövegeiben előforduló hibatípusok és hibák. Az ET jelölés az előtesztet, az UT az utótesztet jelöli. Az ET/UT-val jelölt hibák mindkét szövegben megtalálhatók.

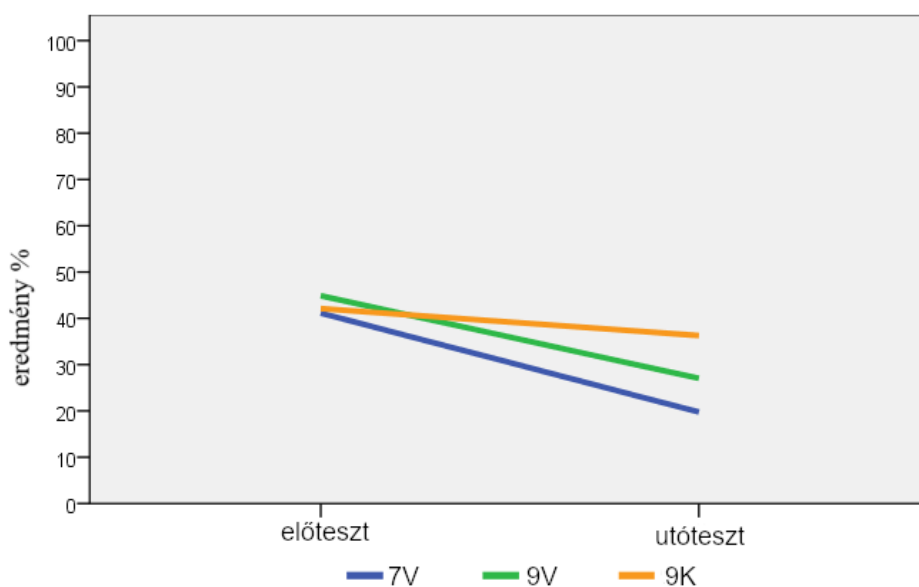
A szövegekben előforduló hibák 3 típushoz tartoznak, elsősorban gyakran előforduló „barkácsolásokat” tartalmaznak. A vizsgálat 3 szakasza – unplugged, semi-unplugged és plugged-in – abban tér el egymástól, hogy mennyire igényel eszköz- (számítógép) használatot. Az unplugged (számítógép nélküli) szakasz biztosít lehetőséget, hogy a tanulók észrevegyék a nyomtatásban is látszó hibákat, ezek elsősorban szintaktikai és tipográfiai hibák. Számítógép használat nélkül korlátozva van a felismerhető hibák száma, így a tanulók munkamemóriája nem telítődik. Amennyiben nincs korlátozás, tehát azonnal a digitális dokumentumban történik a hibakeresés, akkor elsősorban a tördelésben megjelenő hibákat azonosítják a tanulók, valamint túl sok hibát észlelnek egyszerre, így nagy valószínűséggel kevesebb hibát ismernek fel és azonosítanak, mint az általunk alkalmazott szakaszbontásban.

A vizsgálat eredményei közül a párosított tesztek alapján végzett statisztikai elemzés eredményeit emelem ki, amely átfogó képet ad a módszer hatékonyságáról (Sebestyén et al 2021)

	vizsgálati csoport				kontroll csoport	
	7. évfolyam (7V)		9. évfolyam (9V)		9. évfolyam (9K)	
hibatípus	előteszt	utóteszt	előteszt	utóteszt	előteszt	utóteszt
szintaktikai	41,12%	19,75%	44,87%	27,02%	42,11%	36,27%
tipográfiai	0,00%	67,95%	0,00%	60,61%	0,00%	0,88%
tördelési	4,14%	41,73%	30,65%	49,32%	12,55%	36,58%
összesen	19,62%	39,68%	32,73%	45,23%	23,68%	34,74%

12. táblázat Az ERM hatékonyságvizsgálatban résztvevő tanulói csoportok eredményeinek összehasonlítása évfolyamokra és csoportokra bontva.

A tanulási folyamat végére minden csoport magasabb eredményt ért el két hibatípus hibáinak felismerésében és megjelölésében – tipográfiai és tördelési –, valamint összeredmény tekintetében (12. táblázat). A harmadik, szintaktikai hibatípusban minden csoport alacsonyabb százalékban ismerte fel a hibákat az utóteszten az előteszthez képest. A 7V ( $p=0,000$ ) és a 9V ( $P=0,000$ ) csoportokban szignifikáns volt a különbség, míg a 9K ( $p=0,102$ ) csoportban nem (38. ábra).



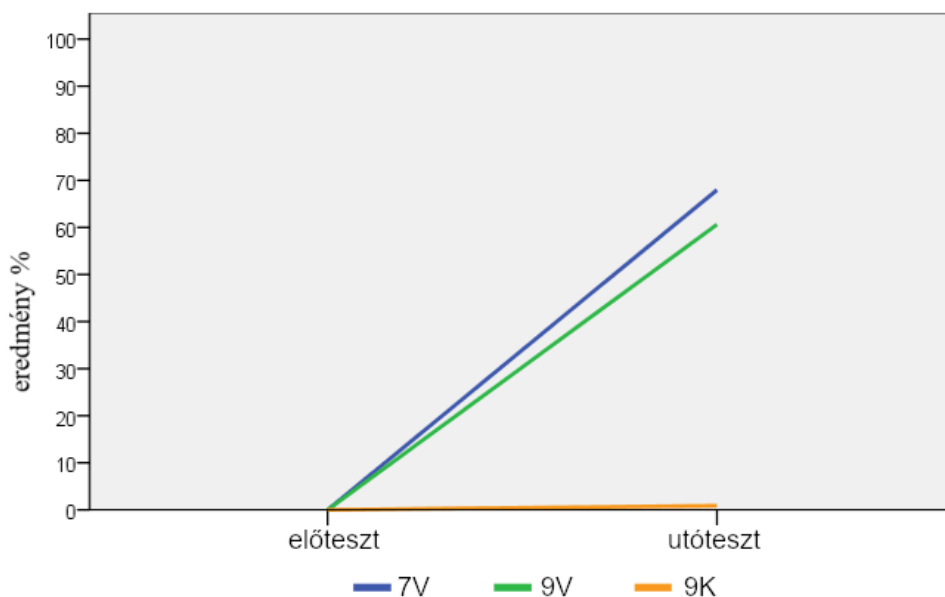
38. ábra A szintaktika hibatípusban elért eredmények és változások az ERM hatékonyságvizsgálat elő- és utótesztje között mindhárom csoportban.

Ez az eredmény elsősorban arra mutat rá, hogy a kontroll csoport csak a szintaktikai hibákra helyezi a hangsúlyt az unplugged szakaszban, figyelmen kívül hagyva a tipográfiai hibákat. Az előteszt és az utóteszt szintaktikai eredményeinek összehasonlítása bebizonyította, hogy a szintaktikai hibák felismerése nagymértékben függ az anyanyelvi képzettségtől. Az előteszt rövidebb szövege nem bővelkedett olyan sok különböző hibával, mint az utóteszté. Az előbbi sokkal jobban megfelelt a 7. évfolyamos tanulók anyanyelvi kompetenciáinak, mint az utóteszt hosszabb szövege. A vizsgálati csoportokban az utóteszt szintaktikai eredményeinek csökkenését az is magyarázza, hogy az előteszten az unplugged szakaszban csak szintaktikai hibákat kerestek – mivel mást nem ismertek –, míg az utótesztben megoszlott a figyelmük a szintaktikai és a tipográfiai hibák között. Ennek a felvetésnek az igazolása további kutatást igényel, azonban azt kijelenthetjük, hogy az informatika órákon a szövegkezelés témakörre fordított órakeret nem elegendő nyelvtani ismeretek átételére. Erre megoldást nyújthat az, ha a nyelvi órákon a tanulók digitális eszközöket használnának a digitális írástudás fejlesztésére és az anyanyelvi tudás integrálásához.

A tipográfiai hibák felismerésében a vizsgálati csoport tagjainak eredménye jelentős javulást mutat, a különbség szignifikáns az előteszt és utóteszt között (7V, 9V  $p=0,000$ ) (39. ábra). A kontroll csoport eredménye stagnál, szignifikáns különbség nem mutatható ki ( $p=0,160$ ).

A csoportok között elsődleges különbség, hogy a kontroll csoport tagjai a tankönyvi feladatokkal, mintákkal találkoztak, ahol az első órákon minden betű- és bekezdésformázást alkalmaztak ugyanazokon a bekezdéseken (Lakosné et al 2019a, 2019b, 2019c). Ez a gyakorlat nem támogatja a tanulók esztétikai érzékének fejlesztését és figyelmen kívül hagyja a tartalom és a formázási stílus kohézióját. A karakterformázások többsége kiemelésre szolgál,

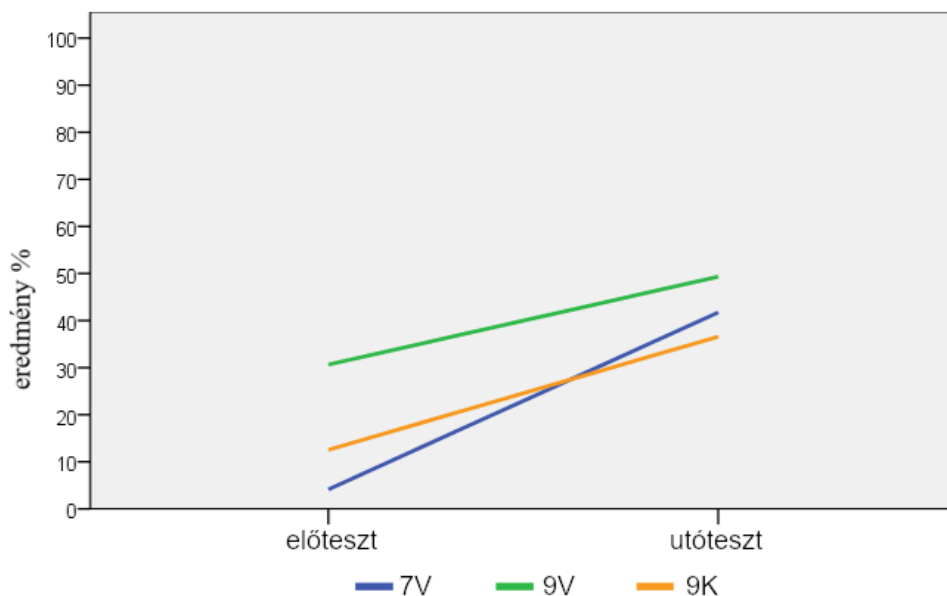
azonban ezek együttes használata, vagy a teljes szövegen való alkalmazása során funkcióját veszti.



39. ábra A tipográfiai hibatípusban elért eredmények és változások az ERM hatékonyságvizsgálat elő- és utótesztje között mindhárom csoportban.

A tördelési hibák esetében mindhárom csoport szignifikáns mértékben jobban teljesít az utóteszten ( $p=0,000$ ) (40. ábra). Azonban nem csak az előteszthez képest szignifikáns az eltérés, hanem a különböző csoportok fejlődésének mértéke, intenzitása között is kimutatható az eltérés. A 9. évfolyamos tanulók fejlődése közel azonos volt, függetlenül attól, hogy mely módszerrel tanultak ( $p=0,173$ ), ezt a 40. ábra is mutatja, ahogy a 9V és 9K csoportok eredményeit szemléltető vonalak közel párhuzamosan futnak. Azonban szignifikáns különbséget találtunk a 7V és a 9V, 9K csoportok között ( $p=0,000$ ;  $p=0,003$ ). Az alacsonyabb, 7. évfolyamos tanulók sokkal nagyobb mértékben fejlődtek. Az ERM tehát már a kezdetektől alkalmazható módszer. A problémamegoldáson, sémaépítésen alapuló szerkezet biztosítja a hosszútávú tudás kialakítását, amit a hagyományos módszerek nem tesznek lehetővé. Ezt iga-

zolja, hogy a 9. évfolyamos tanulók, akik a korábbi évek során több alkalommal tanultak szövegkezelést, az előteszten nem teljesítettek szignifikánsan jobban a 7. évfolyamos tanulóktól (7V és 9K). Az ERM módszer már a szövegkezelés témakör első tanítása során hatékonyabb eredményeket produkál, mint a több éven át tartó, de hagyományos, felületalapú megközelítéssel tanuló csoportok esetében (9K), amely igazolja a [T1] tézist.



40. ábra A tördelési hibatípusban elért eredmények és változások az elő- és az utóteszt között mindhárom csoportban.

A hatékony szövegkezelés oktatásának érdekében szakítani kell a hagyományos, a hivatalos tankönyvek felületi navigációra épülő megközelítéseivel. Fókuszba kell állítani a sémaépítés fejlesztését és a problémamegoldási stratégiák elsajátítását, amellyel a tanulók olyan képességek birtokába kerülnek, amelyeket nem csak az informatika tanórákon tudnak hasznosítani, hanem más tantárgyak esetén, sőt a való életben is. A Hibafelismerés módszere egy példa erre a szemléletváltásra, amely azon túl, hogy tudatosabb számítógéphasználóvá teszi a tanulókat, rámutat arra, hogy mennyire fontos a precizitás, ami

elsőre lehet, hogy több időt vesz igénybe, de hosszútávon megtérül és időt takarít meg.

### 7.3. Spreadsheet Lego (Sprego)

A Sprego módszer hatékonyságvizsgálatát a korábban ismertetett módszerekhez hasonlóan végeztük. Minden esetben középiskolai tanulók szolgálták a minta alapjául, akik a tanulási folyamat előtt és után tesztet írtak (4. számú melléklet). Minden esetben kértünk fel kollégákat, akik a kerettanterv és/vagy tankönyv szerint haladtak, hogy vegyenek részt a felmérésben, így kapva meg a kontroll csoport eredményeit. Több mérésem hiúsult meg a kooperációra felkért pedagógusok visszalépése miatt, így Csapó Gábor publikálta az eredményeket, ahol bizonyította módszer hatékonyságát (Csapó et al. 2020).

Néhány szám és eredmény a hatékonyságvizsgálat eredményeiből. A Csapó által vizsgált minta két 7. évfolyamos (6 osztályos gimnáziumi képzés), valamint egy 10. évfolyamos csoportból állt. Ebből a vizsgálati csoportot alkotta az egyik 7. évfolyamos csoport és a 10. évfolyamos, míg kontroll csoportként a másik 7. évfolyamos csoport vett részt a vizsgálatban (13. táblázat).

	<b>Vizsgálati csoportok</b>		<b>Kontroll csoport</b>
	7. évfolyam	10. évfolyam	7. évfolyam
Tanulók száma	15	18	13
Előteszt	14	16	11
Utóteszt	13	13	13
Párosított tesztek	12	11	11

13. táblázat A Sprego hatékonyságvizsgálatban résztvevő csoportok és tanulók száma. (Csapó et al, 2020)

A vizsgálat során alkalmazott feladatlap (4. számú melléklet) elkészítése során arra törekedtünk, hogy egy fokozatosan nehezedő feladatsort alkossunk, valamint, hogy tartalmát tekintve lefedjük az adatkezelés és programozási ismereteit a táblázatkezelés témakörnek. A feladatokban kitalált függvénynev

(Sprego) és autentikus tartalom szerepel, amely azt a célt szolgálja, hogy felmérjük, hogy ismeretlen adathalmazon hogyan alkalmazzák a tanulók a korábban elsajátított ismereteiket.

Az előzetes tudása a két évfolyamnak szignifikánsan eltérő ( $p=0,0000$ ), hiszen a 7. évfolyam tanulói a Kerettanterv szerint még nem tanultak táblázatkezelést, míg a 10. évfolyamos tanulók már igen. Az évfolyamok közötti különbség a témakör végére eltűnik a vizsgálati csoportban (14. táblázat). Ez a változás arra utal, hogy a módszer igen korai szakaszban is hatékonyan alkalmazható, az algoritmizálási képesség és a sémaépítés fejlesztése nem korfüggő. A két 7. évfolyamos tanulócsoport közötti különbség az utóteszten szignifikánsan eltér a Sprego módszerrel tanult csoport javára, ami alátámasztja a [T1] tézist. A problémamegoldáson, algoritmusépítésen alapuló módszerek hatékonyabbak, mint a felületalapú megközelítések. Ezt támasztja alá Csapó Gábor vizsgálati csoportjainak eredménye, akik az oktatási folyamat végére a táblázatkezelés témakörében szignifikánsan magasabb eredményt produkáltak a kontroll csoporthoz képest ( $p= 0,0013$ ) (14. táblázat).

	vizsgálati csoport		kontroll csoport
	7.	10.	7.
1	94,87%	92,31%	74,36%
2	61,54%	59,62%	42,31%
3	96,15%	93,59%	87,18%
4	79,49%	72,44%	67,31%
5a	60,00%	69,23%	1,54%
5b	40,00%	39,23%	1,54%
5c	42,31%	34,62%	0,00%
5d	39,10%	57,69%	53,21%
5e	76,15%	59,04%	41,35%
5f	68,46%	66,92%	29,23%
5g	61,54%	63,08%	20,00%
Összesen	65,42%	64,34	38,00%

14. táblázat A Sprego hatékonyságvizsgálat utótesztjének eredményei feladat- és csoportbontásban (Csapó et al. 2020).

A módszer hatékonysága mellett a kutatócsoport érdeklődést mutatott a módszerek hosszútávú tudás elérésének mérésére iránt. Elsőként a Sprego módszerrel tanuló csoportok vizsgálatát valósítottuk meg, akik egy évvel a tanulási folyamat után egy késleltetett felmérésben vettek részt.

### 7.3.1. Sprego és a hosszútávú tudás kialakítása

Az eddig ismertetett módszerek – WDC, ERM, Sprego – mindegyike a tanulók algoritmizálási képességének és számítógépes gondolkodásának fejlesztését tűzte ki célul a problémamegoldási stratégiák és a sémaépítés elsajátításával. Mindezeknek köszönhetően elméletileg az átadott tudás és stratégiák a tanulási folyamat alatt a rövid távú memóriából átkerültek a hosszú távú memóriába, azonban a gyakorlatban is szeretnénk volna a módszer hosszútávú hatékonyságát bizonyítani.

A hosszútávú tudás mérésének mintáját 7 csoport alkotta, melyből 4 korábban Sprego módszerrel tanulta a táblázatkezelést, a maradék 3 csoport hagyományos felületalapú módszerekkel (15. táblázat). A vizsgálat résztvevői 8., valamint 10. évfolyam tanulói. Minden csoport egy évvel a tanulási folyamat befejezése után vett részt a vizsgálatban, az eltelt időszakban a tanulók nem tanultak táblázatkezelést.

	vizsgálati csoportok				kontroll csoportok		
	e8_1	e8_2	e10_1	e10_2	c10_1	c10_2	c10_3
N	18	18	17	15	7	13	7

15. táblázat A Sprego késleltetett utóteszten résztvevő csoportok és azok tanulói létszámai.

A csoportok a korábbi évben eltérő óraszámokban tanultak táblázatkezelést a különböző képzési típus (4-6 osztályos gimnázium) miatt. A heti és az összóraszámot a 16. táblázat szemlélteti.

	vizsgálati csoportok				kontroll csoportok		
	e8_1	e8_2	e10_1	e10_2	c10_1	c10_2	c10_3
heti óraszám	1	1	1	1	2	2	1
össz óraszám	8	8	6	6	12	12	6

16. táblázat A késleltetett utóteszten résztvevő csoportok egy évvel korábbi, táblázatkezelés témakörével töltött óraszama.

A kiválasztott csoportok esetében látható alacsony óraszám mögött a magyar oktatási rendszer spirális szerkezete áll, így egy-egy témakör több évfolyamon kerül oktatásra. Az adatok gyűjtése és rögzítése a 2018/2019-es tanév végén valósult meg. A minta összlétszáma 95 fő, melyet a 15. táblázat részletez.

A vizsgálatban használt feladatok a korábbi TAaAS projekt feladatai közül lettek kiválasztva (5. számú melléklet). A feladatok kiválasztásakor a kutatócsoport fókuszában a tanulók adatelemzési, algoritmus- és sémaalkotási képessége, valamint a függvényalkotás előhívása állt (a-e feladatok). Mind ezek mellett a képletkiértékelő és értelmező képességüket is vizsgáltuk az f) feladatban.

- 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 G2!
- e) Írassa ki azon országok számát, amelyek területe nagyobb, mint G2!
- f) Mit csinál az alábbi összetett képlet?  

$$\{=SZUM(HA(B2:B236="Europe";HA(BAL(A2:A236)="A";1)))\}$$

41. ábra A Sprego késleltetett utóteszt feladatai. A hozzátartozó tábla az 5. számú mellékletben található.

Az a) feladatban a tanulóknak a lineáris keresés algoritmusát kell használniuk a legnagyobb ország fővárosának megnevezéséhez. A következő, b) fel-

adatban az ország népsűrűségének kiszámításához a táblázatkezelési ismereteiken túl ismerniük kell a népsűrűség számítási módját, amelyhez matematikai és földrajzi háttérismeret megléte szükséges. Továbbá figyelembe kell venniük a tanulóknak azt is, hogy a lakosság száma ezer főben van megadva. A c) és az e) feladatokban feltételes számlálás algoritmusára van szükség, egyenlőség és egyenlőtlenség esetében, ami a hagyományos módszerrel tanult tanulóknak nehéz lehet felidézni az eltérő szintaktika miatt. A d) feladatban feltételes átlagszámítás található változó használatával, ami a Sprego módszer használatával alig tér el a feltételes összegzéstől és végül az f) feladatban a diákoknak egy komplex tömbképletet kell elemezniük és természetes nyelven leírniuk, hogy mit csinál a képlet.

A javítás során nem tettünk különbséget a megoldás módja szerint, hogy a Sprego módszerből ismert általános függvényeket alkalmazták a tanulók vagy pedig a hagyományos megközelítés által kedvelt problémaspecifikus függvényeket. Elfogadtuk a függvények nevét mind angol, mind magyar nyelven, továbbá akkor is helyesnek rögzítettük az eredményt, ha a függvények neve és argumentumainak sorrendje helyes volt.

	vizsgálati csoport (%)					kontroll csoport (%)				p
	e8_1	e8_2	e10_1	e10_2	együtt	c10_1	c10_2	c10_3	együtt	
a	5,19	6,67	12,94	18,22	10,39	3,81	3,08	0,00	2,47	0,0000
b	20,83	19,44	50,00	23,33	28,31	14,29	32,69	0,00	19,44	0,0841
c	65,00	73,89	62,94	67,33	67,35	21,43	38,72	11,43	27,16	0,0000
d	35,19	59,26	64,71	61,48	54,74	25,40	19,74	0,00	16,09	0,0000
e	48,77	59,26	65,36	56,30	57,35	7,94	20,62	7,94	14,04	0,0000
f	57,41	77,78	74,51	75,56	71,08	14,29	56,41	23,81	37,04	0,0005
Össz.	38,73	49,38	55,08	50,37	48,20	14,52	28,54	7,20	19,37	0,0000

17. táblázat A Sprego készletetett utóteszt eredménye feladat- és csoportbontásban (Csapó et al 2021).

A késleltetett utóteszt eredményeit a 17. táblázat részletezi. A vizsgálati csoportok egy feladat kivételével minden feladatban magasabb eredményt értek el – a b) feladatban a c10\_2 csoport magasabb eredményt produkált, mint az e8\_1 és az e8\_2 csoportok. A teszt összeredményeit figyelembe véve a legmagasabb eredményt az e10\_1 vizsgálati csoport érte el 55,08%-kal, a leggyengébben a kontroll csoport c10\_3 osztálya teljesített. Fontosnak tartjuk kiemelni, hogy az e8\_2 csoport hasonlóan jó eredményeket ért el, mint a vizsgálati csoportban szereplő 10. évfolyamos csoportok, az e10\_1 és az e10\_2, annak ellenére, hogy kétévnyi informatikatanulás van közöttük. Amellett sem lehet elmenni, hogy a vizsgálati csoport leggyengébben teljesítő 8. évfolyamos csoportja (e8\_1) is szignifikánsan magasabb eredményt ért el, mint a tőle 2 évvel idősebb, hagyományos felületalapú módszerekkel tanult csoportjai (c10\_1, c10\_2, c10\_3). A c10\_1 és a c10\_2 csoportok eredményei a magasabb óraszám mellett sem képes megközelíteni a Sprego módszerrel tanított vizsgálati csoport eredményeit.

A Sprego módszer hatékonyságát egyértelműen jelzi a c), d) és e) feladatban elért eredmények, mivel ez a három feladat ugyanazon az algoritmuson alapszik. A vizsgálati csoport szignifikánsan magasabb eredményt ért el mindhárom feladat tekintetében ( $p=0,0000$ ). Ezzel szemben a kontroll csoport tagjai akadályba ütköztek, amikor ugyanazt a feladatot különböző problémáspecifikus függvénnyel – eltérő szintkatikával – próbálták megoldani, a javítás során számos függvénytöredékkel találkoztunk, ahol a tanulók a helyes szintaktikába bonyolódtak bele.

Általánosságban megállapíthatjuk, hogy a Sprego vizsgálati csoportok legrosszabb eredménye is magasabb, a low-mathability megközelítést alkalmazó kontroll csoport legjobbikánál. A Sprego megközelítéssel oktatott tanulók egy évvel a tanulási folyamat befejezése után is magas eredményeket értek el, valamint a kontroll csoporthoz képest összességében szignifikánsan jobb

eredményt produkáltak ( $p=0,000$ ). Feladatokra bontva az eredményeket is egy kivételével – b) feladat – minden esetben szignifikáns az eredmény. Az említett b) feladatban is a vizsgálati csoport tagjai teljesítettek jobban, azonban a Mann-Whitney teszt nem mutatott szignifikáns eltérést a két csoport eredményei között. A javítás során kiderült az is, hogy a csoportok nem azért értek el alacsonyabb eredményt a b) feladatban, mert nem rendelkeznek a képletalkotáshoz szükséges ismeretekkel, hanem mert nem ismerték a népsűrűség kiszámításának mikéntjét. Ez az eredmény rávilágít arra, hogy mennyire fontos (lenne) a tantárgyközi kapcsolatok erősítése.

A magyar közoktatásra jellemző spirális rendszer és a tankönyvek felületalapú megközelítése nem támogatja a hatékony tanulási folyamatokat. A hagyományos, low-mathability módszerek nem fejlesztik a tanulók algoritmizálási és sémaépítési képességeit, így a tananyag nem kerül át a hosszú távú memóriába, így nem érhető el hosszútávú tudás. A témakör ismételt előfordulásakor szinte teljesen az alapoktól kell újrakezdeni, így az egyébként is alacsony óraszám nem elegendő új ismeretek átadására és rögzítésére. Az eredmények egyértelműen bizonyítják, hogy a Sprego sémaalapú high-mathability megközelítésével a tanulók lényegesen jobb eredményt értek el, mint a felületi megközelítésű, low-mathability módszerekkel. A vizsgálat eredményeként megállapítottuk, hogy a Sprego alkalmas, és hatékonyan alkalmazható a hosszútávú tudás kialakítására.

## 8. Összefoglaló

Az informatikaoktatás megérett a paradigmaváltásra, ennek fontosságát már egyre többen felismerik. Tapasztalatok és mérések, azonban egyértelműen igazolják, hogy ezek a törekvések átfogó koncepcionális megfontolások nélkül nem vezetnek, nem vezethetnek gyökeres változásokhoz. A számítógépes és számítógéppel támogatott problémamegoldási és ezen belül az algoritmizálási képességek fejlesztése mint fejlesztési cél, több témakört és tantárgyat érintve is megjelent a 2020-ban bevezetésre kerülő Nemzeti alaptantervben (OFI 2020). Az informatika kerettanterveket (OFI 2008, 2012b, 2020) és a megjelent tankönyveket elemezve (Bánné Mészáros et al. 2008, Bártfai 2011, 2012, Dancsó & Korom 2013, Farkas 2009, Lakosné et al. 2019a, 2019b, 2019c), azonban egyértelmű, hogy a korábbi gyakorlatnak megfelelően, a problémamegoldási és algoritmizálási témakörök mindegyike a programozáshoz köthető. A problémamegoldó szemlélet NAT-szintű támogatása és egy olyan új tartalmi struktúra nem elegendő a változáshoz, amely a gyakorlati megvalósítást és a hozzá szükséges tanári kompetenciákat és azok fejlesztését mellőzi.

A hazai informatikaoktatás a lexikális tudás átadása mellett a minta és leírás (szakácskönyv, computer cooking) alapján történő megvalósítást szorgalmazza a kezdetektől<sup>11</sup> (Csernoch 2021, Papp & Csernoch 2021), és ez a módszer a vizsgáztatásban is a leginkább elterjedt – ECDL (ECDL Foundation 2019), érettségi (Oktatási Hivatal 2021). A módszer hátránya, hogy háttérbe szorítja a szóbeli kommunikációt, valamint a tényleges problémamegoldást. A tanárok többségének a hagyományos, felületközpontú oktatás és a szakács-

---

<sup>11</sup> A dolgozat leadásakor (2021. december), az ötödik osztályos Digitális kultúra tankönyv az első elérhető (Lénárd et al. 2020), azonban az általános iskola többi osztálya számára a tankönyv ezidáig sem készült el.

könyv-típusú feladatok – tankönyvi, ECDL és érettségi feladatsorok – alkalmazása a megszokott, amely szemlélettel szakítani nem, vagy csak nehezen tudnak. A tanulók számára az informatikaórákon a nyelvi és kommunikációs készségek, képességek alkalmazása nehéz, komfortosabb az önálló gyakorlati feladtleírás-követés.

A változást a tanárképzés és a pedagógus továbbképzések reformjában látjuk. A tanárképzés olyan gyökeres átalakítására lenne szükség, ahol képzési időszak lehetőséget biztosítana a tanárjelöltek számítógépes gondolkodásának fejlesztésére, mely nélkülözhetetlen a problémamegoldáson és sémaépítésen alapuló, hosszútávú tudást eredményező módszerek hatékony alkalmazásához. Napjainkban, a közoktatás szereplőinek többsége – szaktól és tudományterülettől függetlenül – nem rendelkezik megfelelő szintű digitális kompetenciákkal, számítógépes gondolkodással ahhoz, hogy hatékonyan alkalmazni tudja az ezen alapuló módszereket, valamint ahhoz, hogy hatékonyan fejleszteni tudja a tanulók számítógépes gondolkodását. Nincs, vagy csak kevés továbbképzés áll a pedagógusok részére, amely felkészítené őket a problémaalapú oktatási megközelítések elsajátítására.

A digitális kompetenciák fejlesztése az egyik kulcsfogalom az Oktatási Hivatal által közzétett tartalmi szabályozókban (OFI 2020), azonban ellentmondva a fejlesztési céloknak, nem fordít figyelmet a dokumentumok helyes szerkesztettségére (OFI 2020). Ez a kettősség is igazolja a számítógépes gondolkodás fejlesztésének fontosságát, továbbá megkérdőjelezi a dokumentum készítőinek rátermettségét, valamint a szabályozók legfontosabb célkitűzését.

A pedagógusok leterheltsége, COVID-19 által hozott kihívások, a minősítési eljárással (EduLine 2020) járó nagymennyiségű dokumentáció elkészítése, háttérbe szorítja a pedagógusok továbbképzések iránti érdeklődését és motiváltságát az önképzésre. 2020 tavaszán, a pandémia miatt elrendelt digi-

tális oktatás hirtelen érkezett, nem volt felkészülve rá országos szinten az oktatási rendszer, azonban a nyári időszakban sem biztosított az Oktatási Hivatal stratégiát a már akkor beharangozott második hullámra. Ez az időszak lehetőséget biztosított volna pedagógusok tömeges továbbképzésére, akár online formában is. Nem csak az informatikatanárok, hanem más pedagógusok digitális kompetenciáinak, számítógépes gondolkodásának fejlesztése vált különösen indokolttá.

Nem került továbbá kidolgozásra, hogy a NAT 2020-ban megemelt informatikaórák – korábbi 5 óra (OFI 2012b) helyett 11 óra (OFI 2020) – ellátására hogyan lesz biztosítható a megfelelő számú és képzettségű pedagógus. Nem történt meg sem a nem-informatika szakos gyakorló tanárok átképzése, továbbképzése, sem a gyakorló informatikusok tanárrá képzése. Nem történt, továbbá változás a nappali szakos tanárképzésben sem, ahol lehetőséget kellene biztosítani az informatika mint harmadik szak felvételére. Mindezen lépések hiányában, az óraszám megemelése önmagában nem lesz elegendő a tanulók számítógépes gondolkodásának fejlesztéséhez.

A kutatásaim kezdeti szakaszában az első éves egyetemi hallgatók alacsony informatikai tudásának hátterében álló okok feltérképezése állt. A kutatásban résztvevő hallgatók mind teljesítették a kerettantervekben foglalt kimeneti követelményeket az általános-, majd a középiskolában, többen szereztek ECDL, valamint informatika érettségi bizonyítványt, melyek a tanulók informatikai tudásának, kompetenciáinak fokmérője. A hozzájuk köthető tanórai felkészülés mégsem teszi lehetővé a hosszútávú és megbízható tudás létrejöttét, nem készíti fel a tanulókat a kor digitális kihívásaira. A digitális írástudás, a számítógépes gondolkodás, az algoritmizálás terén a magyar tanulók átlagon alul teljesítenek, amelyet a PISA mérések (OECD 2011), a TAaAS vizsgálat (Bíró et al. 2015, Csernoch et al. 2015), valamint a mini kompetencia teszt önértékelésen alapuló eredményei igazolnak (Nagy et al. 2021).

A kutatásaim későbbi fókuszában az alkalmazói ismeretek algoritmus-alapú oktatása állt, amelyet a kutatócsoport a programozáshoz hasonlóan a Pólya féle problémamegoldási stratégiára épülve valósít meg. A módszerek építenek a gyors- és lassú gondolkodás megfelelő aktiválásra (Kahneman 2011), valamint a sémaépítésre (Kahneman 2011, Sweller 1988, 1999, 2011, 2019), amelyek együttes alkalmazása lehetővé teszi a hosszútávú tudás kialakítását.

Az elmúlt évek alatt három alkalmazói témakör oktatására fejlesztettünk ki high-mathability (Baranyi & Gilányi 2013, Biró & Csernoch 2015, Borus & Gilányi 2013, Chmielewska et al 2016) megközelítéű módszereket, amelyeket a gyakorlatban is hatékonyan alkalmaztunk. A különböző korosztályú és előképzettségű tanulói csoportokon végzett hatékonyságvizsgálatok egyértelműen mutatják, hogy a magas-mathability, probléma-centrikus módszerek lényegesen hatékonyabbak, mint a széles körben elterjedt low-mathability felületi megközelítések. A fájlkezelés oktatására a webtábla→adattábla fájlkonverzió (WDC – webtable→datatable conversion) megközelítést alkalmaztuk, amely során a tanulók közvetlenül az eredeti interneten elérhető forrással dolgoznak. A kiválasztott források minden esetben a tanulók életkori sajátosságainak és érdeklődési körének megfelel. A webtáblák letöltése és átalakítása során a tanulók természetes nyelven fogalmazzák meg megoldási javaslataikat, amelyeket közösen megbeszélve, algoritmust építve valósítanak meg. A tanulók hozott tudása és ismeretei rendszerezettebbé válnak, valamint megismerik a problémamegoldáshoz kapcsolódó helyes terminológia használatát.

A szövegkezelés oktatása során a Hibafelismerési Modell (ERM – Error Recognition Model) került bevezetésre és tesztelésre. A módszer sajátossága, hogy ugyan szintén a Pólya-féle 4 lépéses stratégiára épül, de nem az első lépéssel kezd, hanem az utolsóval a hibakereséssel, diszkusszióval. Ez a stratégia a programozásoktatásban is bevált és elterjedt. A hibák keresése, kategorizálása és magyarázata rögzíti a helyes dokumentumtervezés és kivitelezéshez

kapcsolódó ismereteket. A tanulók megismerkednek a jólformázott szöveg ismerveivel, valamint törekednek a hibáktól mentes dokumentum készítésére. Ezzel szemben a hagyományos módszerek tankönyvi példái tartalom nélküli szövegeket és az értelmetlen, tipográfiai helytelen formázásokat, valamint a felületi navigációt részesítik előnyben (Dancsó & Korom 2013, Lénárd et al. 2021).

A Sprego (Spreadsheet Lego) táblázatkezelői módszer a formázásokat háttérbe szorítva az algoritmusépítésre, a függvényekre és a képletalkotás folyamatára helyezi a hangsúlyt. Előnye a hagyományos módszerekkel szemben, hogy minimális számú általános célú függvényt használ, amely igény szerint bővíthető, de alapkészlete, ami 12 függvényből áll, lefedi az ECDL és az érettségi követelményrendszerét. A függvények lego módjára egymásba építhetőek, így komplex problémák megoldására is alkalmasak. A módszer hangsúlyt fektet a szövegfüggvényekre, amely a matematikától, számításoktól idegenkedő tanulók, és a fiatalabb korosztályok számára is ideális bevezetésként szolgál.

Mindhárom módszer hatékonysága bizonyítást nyert a hagyományos kerettantervre épülő tankönyvek felületi megközelítést alkalmazó módszereivel szemben, amely igazolta a [T1] hipotézist. A kidolgozott módszerek szorosan kapcsolódnak egymáshoz, minden esetben előkészítik a következő témakört. A fájlkonverzió a szöveg- és a táblázatkezelést, a Sprego pedig a programozást és az adatbáziskezelést. A témakörök szorosan kapcsolódnak a tudástranszfer elemeken keresztül, ami szemben áll a kerettantervek jól elszeparált felépítésével. A témakörök egymást követő tanítása, tapasztalataink szerint fokozottan fejlesztik a tanulók számítógépes gondolkodását. A tanulóknak kialakul egy problémamegoldó szemlélet, amit nem csak az informatikán belül, hanem ezen kívüli akadályok megoldására is tudnak használni. Az azonnali gépi megvalósítást megelőzi a probléma megismerése és elemzése, a tervezés, valamint a

megoldáshoz szükséges eszközök kiválasztása. A tanulók figyelmet fordítanak az eredmény kritikus ellenőrzésére és javítására. A módszerek egymásutánisága, az újra és újra felbukkanó tudáselemek, a tanulók kreativitását igénylő algoritmuslépések, a kutatócsoport tapasztalatai szerint egyértelműen igazolják a [T2] hipotézist, azonban statisztikai adatokkal még nem tudjuk bizonyítani a módszerek szoros kapcsolatából eredő hatékonyságot. Ennek bizonyításához további, éveken átnyúló mérésekre lesz szükségünk, amelyek kivitelezését nagymértékben hátráltatja a jelenlegi járványhelyzet.

A tanulók általános iskolából hozott tudásának mérése során bebizonyosodott, mind a fájlkezelés, mind a szövegkezelés esetében, a tanulók tudása hiányos, rendezetlen, valamint tévhitekkel terhelt, mely igazolta a [T3] hipotézist. A tanulók hozott tudásának eredményeit összehasonlítottuk a mini kompetencia teszt azonos előképzettségű tanulóinak önértékelési eredményeivel. A mini kompetencia teszt résztvevői mind a fájlkezelés, mind a szövegszerkesztés témakörben magabiztosak voltak, tudásukat magas jártassági szinttel azonosították. Az összehasonlítást követően egyértelműen bizonyítást nyert a [T4] hipotézis, hiszen a tanulók hozott tudása és önértékelése szignifikáns mértékben eltér.

Mérési eredményeink egyértelműen bizonyítják, hogy szemléletváltásra van szükség. A kutatócsoport által fejlesztett problémaalapú, sémaépítésre épülő módszerei szakítanak a hagyományos felület- és eszközcentrikus megközelítésekkel. Az informatikatanárok éves konferenciáján is számtalan törekvés és megközelítés kerül bemutatásra, melyekre érdemes nagyobb hangsúlyt fektetni és szélesebb körben terjeszteni. Noha a legtöbbször nem tartozik hatékonyságvizsgálat, hanem egyfajta „legjobb gyakorlat”, ettől függetlenül a módszerek mögött álló koncepció és szemléletmód terjedésére szükség van. A

konferenciákon bemutatott jógyakorlatok és a kutatók együttműködése mindenképpen elengedhetetlen ahhoz, hogy a bemutatott módszerekről bizonyításra kerüljön azok hatékonysága, esetleg elvetése.

A számítógépes gondolkodásra, a digitális írástudásra negyedik alapkészségként tekint több szakirodalmi forrás, így az algoritmizálás és a sémákban való gondolkodás fejlesztése kiemelt szerepet kell, hogy kapjon az oktatási folyamatban, és nem csak a programozásoktatásban.

## 9. Jövőbeli elképzelések

A [T2] tézist nem sikerült egyértelműen bizonyítani, így olyan mérés tervezését és előkészítését tervezem, amely lehetővé teszi a kutatócsoport módszereinek egymásra gyakorolt pozitív hatásának bizonyítását. A kutatás nagyon szigorú ütemtervet igényel, ugyanis egy-egy téma-kör vagy csoport mérésének meghiúsulása az több éves mérési eredményt tesz értékelhetetlenné. A tézis bizonyítását több csoport és intézmény bevonásával tartom megvalósíthatónak, azonban a felmérésben résztvevő pedagógusok – elsősorban a vizsgálati csoportokat tanítók – továbbképzését nélkülözhetetlennek tartom, hogy valóban a kutatócsoport által kidolgozott módszerekkel oktassanak és ne egy kevert megközelítést alkalmazzanak.

A Sprego 2D oktatóprogramhoz több probléma- és függvénybemutatót terveztem, amelyek a módszer függvényeinek egymásba építését szemléltetik és magyarázzák, azonban eddig nem került implementálásra a programba. A vizsgálatok és az oktatóprogram bővítése során – a doktori fokozat megszerzése után is – a kutatócsoporttal szorosan együttműködve szeretnék dolgozni az informatikaoktatás szemléletváltásán.

## 10. Köszönetnyilvánítás

Szeretném köszönetemet kifejezni témavezetőmnek, Csernoch Máriának, aki közel 10 éve támogatja és segíti fejlődésemet és szakmai utamat. Ő a kutatócsoport mozgatórugója, aki szakmai precizitásával és utánozhatatlan munkamoráljával példát mutat a kutatócsoport minden tagjának.

A doktori képzés alatt programvezetőm, Dr. Terdik György professzor úr hasznos publikálási tanácsokkal és javaslatokkal látott el. A doktori iskolának

és az efop pályázatoknak köszönhetően vehettem részt több jeles szakmai konferencián, mind belföldön, mint külföldön, amelyek nem csak publikálási lehetőséget teremtettek számomra, hanem nemzetközi kapcsolatok kötését, és magas színvonalú szakmai vitákat tettek lehetővé.

Köszönetet szeretnék mondani továbbá tanárainknak, kollégáimnak, akiknek részvétele nélkül nem sikerült volna az oktatási módszerek hatékonyságának bizonyítása. Köszönet illeti a Móricz Zsigmond Református Kollégium, Gimnázium, Szakgimnázium és Általános Iskolát, amely lehetőséget biztosított új módszerek tesztelésére és alkalmazására, valamint a vizsgálatokban résztvevő, minden tudásukat beleadó tanulókat.

Kiemelt köszönettel és hálával tartozom Csapó Gábornak, kutatótársamnak, akivel 10 éve barátság és 7 éve közös szakmai törekvések kötnek össze. Az elmúlt évek során mindig szorosán együttműködve, egymást támogatva és segítve dolgoztunk és remélem, hogy a jövőben is lesz erre példa.

Családom feltétel nélküli támogatásáért, odaadásáért és buzdításáért igazán hálás vagyok, tekintettel voltak szakmai elköteleződésemmre, feladataimra, lehetőséget és időt biztosítottak szakmai ambícióim, törekvéseim megvalósításához.

## 11. Summary

Based on various competence tests, it is obvious that it is high time for a paradigm shift in the Hungarian ICT education. In the competence tests Hungarian students perform below the average, which is confirmed by the results of the PISA measurements (OECD 2011), the TAaAS tests (Biró et al. 2015) and the mini-competence tests administered by our research group. Although the National Base Curriculum emphasizes the importance of the development of students' computational thinking, computer problem solving, and algorithmic skills and abilities, the frame curricula narrows it down to programming. A new point of view as well as a new structure of the syllabus is not enough to be changed, because it ignores the implementation in practice. There is not any or there are only few postgraduate trainings available for teachers which prepare (provide?) them master level problem-based educational strategies. The sudden pandemic in 2020 found the national system unprepared for it at national level. This period should provide an opportunity for speeding up, as well as widening in-service training of educators, either online or personally. In these trainings both ICT and non-ICT teachers should participate so as to develop their digital competencies and their computational thinking abilities, as well.

My research focuses on the didactical questions of teaching non-traditional programming subject with methods adapted from programming. These methods are conscious and proved to be effective and efficient in teaching programming. These are primarily based on Pólya's concept-based problem-solving approach with the enquiry-based teaching method. The adapted Pólya methods are proved to be effective with the right proportion of fast and slow thinking, the wide range of guidance arriving from teachers (Kirschner 2002),

as well as from schema building that allows the development of long-term knowledge.

In recent years, our research group has developed high-mathability methods so as to teach three computer application topics, which compared to the widely accepted and applied traditional surface approach methods in the learning-teaching process. In teaching file management we use the webtable→datatable conversion (WDC) approach, where students work directly with original web-based data sources and apply a conversion process to reach normalized data tables. In each case, the sources are chosen in accordance with the age, characteristics, background knowledge and interests of the students. One of the fundamental sections of the conversion process is building algorithms. It is expressed in the students' mother tongue, but is clearly separated from the algorithm itself, actually it is the selection of the tool for the effective execution of the task. All these processes are carried out with full guidance of teachers as it is suggested by Kirschner et al. (2006). Learners' knowledge and skills become more systematic and they learn to use the correct problem solving related terminology.

The Error Recognition Model (ERM) is introduced in text management. Although ERM is also based on Pólya's 4-step strategy, its peculiarity is that it does not start with the first step, but with the last one—debugging and discussion. This approach has also been proven to be effective, consequently, widely accepted in programming education. Recognizing, correcting, and explaining errors build up such a knowledge which is related to correct document design and formatting. Students become familiar with the criteria of a properly-formatted text and oriented to create a document without errors. In contrast, traditional textbooks prefer decontextualized and meaningless materials, typographically incorrect formatting and surface navigation.

The Spreadsheet Lego (Sprego) approach focuses on real world problem-solving, building up algorithms and coding in the simplified functional language of spreadsheets. Its advantage over traditional methods is that it uses a minimum number of general-purpose functions that can be expanded on demand, but its basic set of 12 functions covers the ECDL and graduation requirements. Due to the limited number of functions, this method highly supports the building of composite functions. With the application of n-ary as well as composite functions, teaching spread sheeting with Sprego strengthens the concept of function originated in mathematical studies. Another characteristic of this method is that instead of pure Math problems, it emphasizes the application of such text-based problems which improve students' familiarity.

The effectiveness of all three methods (WDC, ERM and Sprego) has been proven compared to the widely accepted surface approach methods, which are based on the traditional framework curricula and which confirm the [T1] thesis. The methods detailed in the present dissertation are closely related to each other. The WDC method combines text- and spreadsheet management, the ERM method is primarily about text-management, but closely related to the WDC method, while Sprego is the beneficial of both WDC and ERM. Further role of Sprego is to provide firm background for further studies in ICT, especially in database-management and programming. The topics are closely related to the activation of knowledge transfer elements and schema-building. However, we cannot prove the [T2] thesis with quantitative data, our educational experience clearly indicates the advantages of the three combined methods. To prove of the effectiveness of the activation of knowledge-transfer requires further studies.

In measuring students' knowledge brought from primary school both in file management and word processing, we found that students' knowledge was unreliable, disordered and burdened with misconceptions which confirm the

[T3] thesis. The first-year students of secondary education cannot or can hardly rely on the knowledge they have brought from primary school. The level of students' knowledge they brought had been compared to the students' level of self-evaluation. We found that the students tested showed confidence in both file- and text-management, declaring high level of proficiency in both subjects. After the comparison of self-evaluation values to students' results in a mini-competence test, the [T4] thesis has clearly been proven, as the knowledge and self-evaluation of showed a significant difference.

The results of the measurements of the research group clearly demonstrate that there is a need for making changes in approach. It is clearly declared that computational thinking skills should be the fourth fundamental skill along with the 3Rs (Reading, wRiting, aRithmetic) (Wing, 2006). So as to develop students' computational thinking skills, ICT education must open up end-user computing, which was suggested almost thirty years ago by Soloway and his fellows (1993), and should not be restricted to programming education, exclusively.

## 12. Irodalomjegyzék

- Anderson, J. R. (1993). Problem solving and learning. *American Psychologist*, 48, pp. 35–44.
- Abraham, R., Erwig, M. (2009). Mutation Operators for Spreadsheets. *Software Engineering, IEEE Transactions on Volume: 35, Issue: 1.* pp 1–17.
- Adaboh, S. (2016). Using the cognitive load theory to assist in the design of instruction for the university lecture room: some key lessons. *Asian Journal of Education Research*, 4., pp. 53–59.
- Bánné Mászáros, A., Csintalan, T. & Lakosné Makár, E. (2008). *Informatika 5. Nemzeti Tankönyvkiadó*
- Baranyi, P. & Gilányi, A. (2013). Mathability: Emulating and Enhancing Human Mathematical Capabilities. 2013 IEEE 4th International Conference on Cognitive Infocommunications (CogInfoCom), pp. 555–558.
- Bártfai, B. (2011). *Számítógéphasználat mindenkinek. BBS-INFO Könyvkiadó és Informatikai Kft., Budapest.*
- Bártfai, B. (2012). *Excel haladóknak. BBS-INFO Könyvkiadó és Informatikai Kft., Budapest.*
- Báthory, Z. (1997). *Tanulók, iskolák – különbségek: Egy differenciált tanításmélet vázlata. OKKER Kiadó, Budapest.*
- Bell, T. & Newton, H. (2013). *Unplugging Computer Science. Improving Computer Science Education. Routledge.* pp. 68–81.
- Ben-Ari, M. (1999). *Bricolage Forever!, PPIG 1999. 11th Annual Workshop. 5–7 January 1999, Computer-Based Learning Unit, University of Leeds.* pp. 1-5. Letöltve 2020.10.10. <https://ppig.org/files/1999-PPIG-11th-benari.pdf>
- Ben-Ari, M. & Yeshno, T. (2006). Conceptual models of software artifacts. *Interacting with Computers*. 2006, 18(6), pp. 1336–1350.
- Bernát, P. (2015). *Robotika az általános iskolában és a RoboMind programozási környezet. InfoDidact'15. 10. Informatika Szakmódszertani Konferencia. Webdidaktika Alapítvány*
- Bernát, P. & Zsakó, L. (2019). *Programozás tanítási módszerek – stratégia a kezdetekre. InfoDidact'19. 12. Informatika Szakmódszertani Konferencia. Webdidaktika Alapítvány*

- Biró, P. & Csernoch, M. (2015). The mathability of computer problem solving approaches. 6th IEEE Conference on Cognitive Infocommunications CogInfoCom 2015: Proceedings, October 19-21, Széchenyi István University Győr, pp. 111–114.
- Biró, P., Csernoch, M., Máth, J. & Abari, K. (2015). Testing Algorithmic Skills in Traditional and NonTraditional Programming Environments. Informatics in Education, Vilnius University, Vol. 14, No. 2., pp. 175–197.
- Biró, P., Csernoch, M., Sebestyén, K. & Eszter Gombos (2018). Algorithmic Skills Vs. Time Spent On Computers And Mobile Devices. Turkish Online Journal of Educational Technology, INTE 2018 November, pp. 133–140. Retrieved 2020.05.05. from [http://www.tojet.net/special/2018\\_12\\_2.pdf](http://www.tojet.net/special/2018_12_2.pdf)
- Borus, G. Gy. & Gilányi, A. (2013). Solving systems of linear functional equations with computer. 2013 IEEE 4th International Conference on Cognitive Infocommunications. pp. 559–562.
- Carr, N. (2011). The shallows: What the Internet is doing to our brains. New York: WW Norton and Company.
- Chmielewska, K., Gilányi, A. & Łukasiewicz, A. (2016). Mathability and Mathematical Cognition. 7th IEEE International Conference on Cognitive Infocommunications, CogInfoCom.
- Csapó, G. & Sebestyén, K. (2015). Oktatóprogram a Sprego táblázatkezelő módszerhez. InfoDidact 2015, 8. Informatika Szakmódszertani Konferencia. Webdidaktika Alapítvány. Letöltve 2020.02.12. <http://people.inf.elte.hu/szlavi/InfoDidact15/Manuscripts/CsGSK.pdf>
- Csapó, G. & Sebestyén, K. (2017). Educational Software for the Sprego Method. The Turkish Online Journal of Educational Technology, 2017 október, pp. 987–1000. Letöltve 2020.06.12. from [http://www.tojet.net/special/2017\\_10\\_1.pdf](http://www.tojet.net/special/2017_10_1.pdf)
- Csapó, G., Csernoch, M. & Abari, K. (2020). Sprego: case study on the effectiveness of teaching spreadsheet management with schema construction. Education and Information Technologies 25, pp. 1585–1605.
- Csapó, G., Sebestyén, K., Csernoch, M. & Abari, K. (2021). Case study: developing long-term knowledge with Sprego. Education and Information Technologies 26, pp. 965–982.
- Csernoch M. (2009). Teaching word processing – the theory behind, Teaching Mathematics and Computer Science, 2009/1, pp. 119–137.
- Csernoch M. (2010). Teaching word processing – the practice, Teaching Mathematics and Computer Science, 2010, (8)2, pp. 247–262.

- Csernoch M. (2011). Clearing Up Misconceptions About Teaching Text Editing. ICERI2011: 4th International Conference of Education, Research and Innovation. Madrid, Spain. pp. 407–415.
- Csernoch M. (2016). Algorithms and Schemata in Teaching Informatics. Debreceni Egyetemi Kiadó, Debrecen. Letöltve 2019.01.25. [http://tanarkepzes.unideb.hu/szaktarnet/kiadvanyok/algorithmusok\\_es\\_semak\\_2.pdf](http://tanarkepzes.unideb.hu/szaktarnet/kiadvanyok/algorithmusok_es_semak_2.pdf)
- Csernoch, M. (2017). Thinking Fast and Slow in Computer Problem Solving. Journal of Software Engineering and Applications, Vol. 10. No 1., 2017. pp. 11–40. Letöltve 2020.10.25. [http://file.scirp.org/pdf/JSEA\\_2017012315324696.pdf](http://file.scirp.org/pdf/JSEA_2017012315324696.pdf)
- Csernoch M. (2019). Do You Speak and Write in Informatics? The 10th International Multi-Conference on Complexity, Informatics and Cybernetics, March 12–15, 2019, Orlando, Florida, USA. pp.147–152.
- Csernoch, M. (2021). Digitális kultúra – Prezentáció – Tudástranszfer. Infodidact 2021 Informatika Szakmódszertani Konferencia.
- Csernoch, M. & Biró, P. (2015a). Számítógépes problémamegoldás. Tudományos és Műszaki Tájékoztatás 62. évf. 3. szám. pp. 86–94.
- Csernoch, M. & Biró, P. (2015b). Wasting Human and Computer Resources. International Journal of Social, Education, Economics and Management Engineering, 9(2). pp. 573–581.
- Csernoch, M. & Biró, P. (2019). Are digital natives spreadsheet natives? Proceedings of the EuSPRIG 2019 Conference "Spreadsheet Risk Management", Browns, Covent Garden, London. pp. 1–12.
- Csernoch, M., Biró, P., Máth, J. & Abari, K. (2015). Testing algorithmic skills in traditional and non-traditional programming environments. Informatics in Education, 2015, Vol. 14, No. 2. pp. 175–197.
- Csernoch, M. & Dani, E. (2017). Data-structure validator: An application of the HY-DE model. 2017 8th IEEE International Conference on Cognitive Infocommunications (CogInfoCom). Debrecen.
- Csókás A. (2019). Százegres is lehet a fizetésemelés. Magyar Nemzet. 2019.11.02.
- Dancsó, T. & Korom, P. (2013). Informatika 9-10. – A gimnáziumok számára. Nemzedékek Tudása Tankönyvkiadó. Budapest.
- De Groot, A. D. (1956). Über das Denken des Schachspielers. Rivista di psicologia. 50. pp. 73–104.

- Dercsényi, D. (2016). Kiakadt az elavult számítésh-tankönyvön? Az ügy háttérén még jobban ki fog. Letöltve: 2020.10.11. [https://hvg.hu/itthon/20160920\\_elavult\\_informatika\\_konyv\\_mozaik\\_kiado](https://hvg.hu/itthon/20160920_elavult_informatika_konyv_mozaik_kiado)
- Dewey, J. (1910). *How we think*. Holt, Rinehart and Winston, New York.
- ECDL Foundation (2019). ECDL vizsgapédatár. Számítógésh-alapismeretek. Syllabus 1.0. Neumann János Számítógésh-tudományi Társaság.
- EduLine (2020). Az új minősítési rendszertől függ majd a tanárok fizetése. EduLine. Letöltve: 2020.05.06. [https://eduline.hu/kozoktatás/20200130\\_Az\\_új\\_minosites\\_i\\_rendszertol\\_fugg\\_majd\\_a\\_tanarok\\_fizetese](https://eduline.hu/kozoktatás/20200130_Az_új_minosites_i_rendszertol_fugg_majd_a_tanarok_fizetese)
- Európai Bizottság (2020). Bizottsági szolgálati dokumentum. 2020. évi országjelentés – Magyarország. Letöltve: 2020.12.05. [https://ec.europa.eu/info/sites/info/files/2020-european\\_semester\\_country-report-hungary\\_hu.pdf?fbclid=IwAR3\\_bVQD\\_rA14mK5jNk-ZenPULUFmq6d2mIgAKqZpX6qD0iQ7tLa1nr60nvg](https://ec.europa.eu/info/sites/info/files/2020-european_semester_country-report-hungary_hu.pdf?fbclid=IwAR3_bVQD_rA14mK5jNk-ZenPULUFmq6d2mIgAKqZpX6qD0iQ7tLa1nr60nvg)
- EuSpRIG (2020). Horror Stories. Letöltve: 2020.01.02. <http://eusprig.org/horror-stories.htm>
- Farkas, Cs. (2009). Informatikai ismeretek a 9-10. évfolyam részére. Jedlik Oktatási Stúdió, Budapest.
- Freiermuth, K.; Hromkovic, J. & Steffen, B. (2008). *Creating and Testing Textbooks for Secondary Schools. Informatics Education - Supporting Computational Thinking*. ISSEP 2008. Lecture Notes in Computer Science, vol 5090. Springer, Berlin, Heidelberg.
- Gander, W. (2014). *Informatics and General Education. Informatics in Schools. Teaching and Learning Perspectives*. ISSEP 2014. Lecture Notes in Computer Science, vol 8730. Springer, Cham.
- Garrett, N. (2015). Textbooks for Responsible Data Analysis in Excel. *Journal of Education for Business* 90(4), pp. 169–174.
- Gould, J. (1975). Some psychological evidence on how people debug computer programs. *International Journal of Man-Machine Studies*, (7) 1, pp. 151–182.
- Hattie, J. (2003), *Teachers Make a Difference: What is the research evidence? Distinguishing Expert Teachers from Novice and Experienced Teachers*. Australian Council for Educational Research (ACER) Annual Conference on: Building Teacher Quality. Letöltve 2020.05.06. [https://www.det.nsw.edu.au/proflearn/docs/pdf/qt\\_hattie.pdf](https://www.det.nsw.edu.au/proflearn/docs/pdf/qt_hattie.pdf).
- Hattie, J. (2012), *Visible Learning for Teachers: Maximizing Impact on Learning*. Routledge.
- Hermans, F. (2019). How to teach programming (and other things)?. Letöltve 2020.12.10. <https://www.youtube.com/watch?v=g1ib43q3uXQ>

- Hutchinson, E. D. (1949). *How to think creatively*. Abingdon Cokesbury, New York.
- Hubwieser, P. (2004) *Functional Modeling in Secondary Schools using Spread-sheets*. *Education and Information Technologies*, 9(2), pp. 175–183.
- IEEE&ACM Report 2013 (2013). *Computer Science Curricula 2013. Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. December 20, 2013. The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) IEEE Computer Society. Letöltve 2020.12.10. [https://www.acm.org/binaries/content/assets/education/cs2013\\_web\\_final.pdf](https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf)
- IMDb (2019). *Top Rated Movies*. Letöltve 2020.10.08. <https://www.imdb.com/chart/top>
- Jerinic, L. (2014). *Teaching Introductory Programming - Agent-based Approach with Pedagogical Patterns for Learning by Mistake*. (IJACSA) *International Journal of Advanced Computer Science and Applications* Vol. 5, No. 6, 2014., pp. 60–69.
- Johnson, D. M. (1972). *Systematic introduction to the psychology of thinking*. Harper and Row, New York.
- Jorgensen, H. (2013). *How not to Excel in economics. The interpreter*. Lowy Institute. Letöltve 2018.07.07. <http://www.lowyinterpreter.org/post/2013/04/18/How-not-toExcel-in-economics.aspx>
- Jukes, I. & Dosaj, A. (2006). *Understanding Digital Children (DKs). Teaching & Learning in the New Digital Landscape*. The InfoSavvy Group. Prepared for the Singapore MOE Mass Lecture
- Jukes, I. & McCain, T. & Crockett, L. (2010). *Understanding the Digital Generation*. SAGE Publications. Thousand Oaks, United States.
- Jury, D. (2004). *About Face: Reviving The Rules of Typography*. RotoVision, SA, Switzerland.
- Jury, D. (2006). *What is Typography?*. RotoVision, SA, Switzerland.
- Kadijevich, D. (2009). *Simple spreadsheet modeling by first-year business undergraduate students: Difficulties in the transition from real world problem statement to mathematical model*. *Mathematical applications and modeling in the teaching and learning of mathematics: Proceedings the 11th International Congress on mathematical Education, Mexico*, pp. 241–248.
- Kadijevich, D. (2013). *Learning about spreadsheet. Improving Computer Science Education*. New York and London: Routledge; pp. 19–33.
- Kahneman, D. (2011). *Thinking, Fast and Slow*. New York: Farrar, Straus and Giroux.

- Kátai, Z., Osztián, E., & Vekov, G. K. (2016). Promoting computational thinking by artistically enhanced algorithm visualization. *InfoDidact 2015.*, 10. Informatika Szakmódszertani Konferencia. Webdidaktika Alapítvány. Letöltve: 2020.10.17.  
<https://people.inf.elte.hu/szlavi/InfoDidact16/Manuscripts/KZOEVGK.pdf>
- Kerekes, N. (2013). Tehetséges fiatalok motivációjáról és szocializációs, társas jellemzőiről. *Vzdelávanie, výskum a metodológia (Oktatás, kutatás és módszertan)*. International Research Institute s.r.o., Komárno. pp. 361–370.
- Kirschner, P. (2002). Cognitive load theory: Implications of cognitive load theory on the design of learning. *Learning and Instruction*. 12. pp. 1–10.
- Kirschner, P., Sweller, J. & Clark, R.E. (2006). Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist Discovery. *Problem-Based, Experiential, and InquiryBased Teaching*. *Educational Psychologist*, 41(2), pp. 75–86.
- Kirschner, P. & De Bruyckere, P. (2017). The myths of the digital native and the multi-tasker, *Teaching and Teacher Education*. 67 (2017), pp. 135–142.
- Klebelberg Központ (2020). Klebelberg Képzési Ösztöndíj Program. Letöltve: 2020.12.10.  
<https://kk.gov.hu/klebelberg-kepzesi-osztondij>
- Kozéki B. (1980). A motiválás és motiváció összefüggéseinek pedagógiaipszichológiai vizsgálata. Akadémiai Kiadó. Budapest.
- Kruger, J. & Dunning, D. (1999). 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, 77 (6), pp. 1121–1134.
- Krugman, P. (2013). The excel depression. *The New York Times*, April 19, 2013.
- Kwak, J. (2013). The Importance of Excel. The Baseline Scenario. Letöltve 2020.10.12.  
<http://baselinescenario.com/2013/02/09/the-importance-of-excel>
- Lakosné Makár, E., Rajk, Á., Regele, Gy. & Ridzi, G. (2019a). Informatika 6., OFI. Letöltve 2020.10.01. [https://www.tankonyvkatalogus.hu/pdf/NT-11682\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/NT-11682__teljes.pdf)
- Lakosné Makár, E., Rajk, Á., Regele, Gy. & Ridzi, G. (2019b). Informatika 7., OFI. Letöltve 2020.10.01. [https://www.tankonyvkatalogus.hu/pdf/NT-11782\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/NT-11782__teljes.pdf)
- Lakosné Makár, E., Rajk, Á., Regele, Gy. & Ridzi, G. (2019c). Informatika 8., OFI. Letöltve 2020.10.01. [https://www.tankonyvkatalogus.hu/pdf/NT-11882\\_\\_teljes.pdf](https://www.tankonyvkatalogus.hu/pdf/NT-11882__teljes.pdf)
- Lénárd F. (1978). A problémamegoldó gondolkodás. Akadémiai Kiadó, Budapest
- Lénárd, A., Abonyi-Tóth, A., Turzó-Sovák, N., Varga, P. (2020). *Digitális kultúra tankönyv* 5. Oktatási Hivatal. Budapest.

- Merriënboer, J.J.G. van & Sweller, J. (2005). Cognitive Load Theory and Complex Learning: Recent Developments and Future Directions. *Educational Psychology Review*, 2015, 17(2), pp. 147–177.
- Microsoft (2020). A billentyűzet használata. Letöltve: 2020.10.10. <https://support.microsoft.com/hu-hu/windows/a-billentyűzet-használata-18b2efc1-9e32-ba5a-0896-676f9f3b994f>
- Microsoft (2020). DARABTELI függvény. Letöltve 2020.11.11. <https://support.microsoft.com/hu-hu/office/darabte-li-f%C3%BCggv%C3%A9ny-e0de10c6-f885-4e71-abb4-1f464816df34>
- Nagy, T. K., Csernoch, M. & Biró, P. (2021). The Comparison of Students' Self-Assessment, Gender, and Programming-Oriented Spreadsheet Skills. *Education Sciences* 2021, 11(10), 590. pp. 1–29.
- Nagy, S. (1981). *Az oktatásemélet alapkérdései*. Tankönyvkiadó. Budapest
- National Research Council (2000). *How people learn: Brain, mind, experience, & school: Expanded edition*. National Academies Press. Washington.
- Newell, A., Shaw, J. C. & Simon, H. A. (1962). *The process of creative thinking*. Contemporary approaches to creative thinking. Atherton Press, New York.
- OECD (2011). *PISA 2009 Results: Student On Line: digital technologies and performance (Volume VI)*. Letöltve 2020.06.10. <https://www.oecd.org/pisa/pisaproducts/48270093.pdf>
- OECD (2015). *Main Results from the PISA 2012 Computer-Based Assessments, in Students, Computers and Learning: Making the Connection*, OECD Publishing, Paris. Letöltve 2020.02.10. <https://www.oecd.org/pisa/pisaproducts/PISA-2012-technical-report-final.pdf>
- OFI (2008). *Kerettanterv. 2/2008. (II.8.) OKM rendelete a kerettantervek kiadásának és jóváhagyásának rendjéről*. Letöltve: 2020.06.10. <http://www.nefmi.gov.hu/kozoktatas/tantervek/oktatasi-kulturalis>
- OFI (1995). *NAT 1995. 130/1995. (X.26). Korm. rendelet a Nemzeti alaptanterv kiadásáról*. Letöltve: 2020.03.04. [http://njt.hu/cgi\\_bin/njt\\_doc.cgi?docid=24382.38666](http://njt.hu/cgi_bin/njt_doc.cgi?docid=24382.38666)
- OFI (2012a). *NAT 2012. 110/2012. (VI. 4.) Korm. rendelete a Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról*. Letöltve: 2020.03.10. from [http://ofi.hu/sites/default/files/attachments/mk\\_nat\\_20121.pdf](http://ofi.hu/sites/default/files/attachments/mk_nat_20121.pdf)

- OFI (2012b). Kerettanterv. 51/2012. (XII. 21.) számú EMMI rendelet – a kerettantervek kiadásának és jóváhagyásának rendjéről. Letöltve: 2020.06.12. from <http://kerettanterv.ofi.hu/>
- OFI (2020). NAT 2020. 5/2020. (I.31.) Korm. rendelet: .A Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról szóló 110/2012. (VI.4.) Korm. rendelet módosításáról. Letöltve: 2021.01.06. <https://magyarkozlony.hu/dokumentumok/3288b6548a740b9c8daf918a399a0bed1985db0f/megtekintes>
- Oktatási Hivatal (2021). Központi írásbeli feladatsorok, javítási-értékelési útmutatók. Feladatsorok és javítási-értékelési útmutatók a közép- és emelt szintű írásbeli érettségi vizsgákon - 2005-től napjainkig. Letöltve 2021.08.10. <https://www.oktatas.hu/koznevels/erettsegi/feladatsorok>
- Owen, E. & Sweller, J. (1989). Should problem-solving be used as a learning device in mathematics? *Journal for Research in Mathematics Education*, 20. 3. sz. pp. 322–328.
- Osztíán, E., Kátai, Z., & Vekov, G. K. (2017). Multi-Dimensional Expansion of Algorhythms. *The Turkish Online Journal of Educational Technology, INTE* 2017 November, pp. 573–578.
- Panko, R. R. (2008). What we know about spreadsheet errors. *Journal of End User Computing's Special issue on Scaling Up End User Development Volume 10, No 2. Spring 1998*, pp. 15–21.
- Panko, R. R. & Aurigemma, S. (2010). Revising the Panko-Halverson taxonomy of spreadsheet errors. *Decis. Support Syst.* 2015. 49, 2. pp. 235–244.
- Panko, R. R. & Port, D. (2013). End User Computing: The Dark Matter (and Dark Energy) of Corporate It. *Journal of Organizational and End User Computing*, 2013, 25 (3), pp. 1–19.
- Papp, G. & Csernoch, M. (2021). PowerPoint prezentációk elemzése négy korosztályban. *Infodidact 2021 Informatika Szakmódszertani Konferencia*.
- Pénzcentrum (2020). Mégsem lesz 30 százalékos béremelés az oktatásban? Megkapták a bérajánlatot a tanárok. Pénzcentrum. Letöltve: 2020.07.05. <https://www.penzcentrum.hu/karrier/megsem-lesz-30-szazalekos-beremeles-az-oktatasban-megkaptak-a-berajanlatot-a-tanarok.1096908.html>
- Pólya, G. (1954). *How To Solve It. A New Aspect of Mathematical Method*. Princeton University Press, Princeton, New Jersey.
- Powell, S. G., Baker, K. R. & Lawson, B. (2008). A critical review of the literature on spreadsheet errors. *Decision Support Systems*, 46(1), pp. 128–138.

- Powell, S. G., Baker, K. R. & Lawson, B. (2009a). Errors in Operational Spreadsheets. *Journal of Organizational and End User Computing*, 21(3), pp. 24–36.
- Powell, S. G., Baker, K. R. & Lawson, B. (2009b). Impact of errors in operational spreadsheets. *Decision Support Systems*, 47(2), pp. 126–132.
- Prensky, M. (2001). *Digital Natives, Digital Immigrants - From On the Horizon*. MCB University Press, Vol. 9 No. 5, October 2001., pp. 1–6.
- Reynolds, G. (2008). *Presentation Zen: Simple Ideas on Presentation Design and Delivery*. Pearson Education Inc. Berkely.
- Sebestyén, K. (2014). Te is lehetsz informatikatanár (?). *InfoDidact 2014*. 9. Informatika Szakmódszertani Konferencia. Webdidaktika Alapítvány Letöltve: 2020.10.01. <http://people.inf.elte.hu/szlavi/InfoDidact14/Manuscripts/SK.pdf>
- Sebestyén, K. (2020). Students' Knowledge in File Management. *Proceedings of the 11th International Conference on Applied Informatics*, pp. 296–308.
- Sebestyén, K. & Csapó, G. (2018). Visualising Sprego Inequality Problems With 2d Representations. *Turkish Online Journal of Educational Technology*, INTE 2018 November, pp. 1081–1091. Letöltve 2020.05.05. [https://www.int-e.net/publication\\_folder/inte/inte\\_iticam\\_idec2018\\_v2.pdf](https://www.int-e.net/publication_folder/inte/inte_iticam_idec2018_v2.pdf)
- Sebestyén, K., Csapó, G. & Mária Csernoch (2019). Introduction to Algorithmic Based Data Management in Spreadsheet Environment. *The Turkish Online Journal of Educational Technology*. INTE 2019, pp. 354-361. Letöltve 2020.06.08. [https://www.int-e.net/publication\\_folder/inte/inte\\_iticam\\_2019.pdf](https://www.int-e.net/publication_folder/inte/inte_iticam_2019.pdf)
- Sebestyén, K., Csapó, G., Csernoch, M. & Aradi, B. (2021) Error Recognition Model: End-user Text Management. *Acta Polytechnica Hungarica*. Közlésre elfogadott.
- Sestoft, P. (2011). *Spreadsheet technology*. IT University Technical Report Series, IT University of Copenhagen. pp. 20.
- Skemp, R. (1971). *The Psychology of Learning Mathematics*. Lawrence Erlbaum Associates. New Jersey.
- Soloway, E. (1993). Should we teach students to program?. *Communications of the ACM*, 36(10). pp. 21–24.
- Sternberg, R. J. (1980). Sketch of a componential subtheory of human intelligence. *The Behavioral and Brain Sciences*, 3. pp. 573–614.
- Sweller, J.(1988). Cognitive load during problem solving: Effects on learning, *Cognitive Science*, 12, pp. 257–285.

- Sweller, J. (1999). Instructional Design in Technical Areas, *Australian Education Review*, No. 43.
- Sweller, J., Ayres, P. & Kalyuga, S. (2011). *Cognitive Load Theory*. Springer New York Dordrecht Heidelberg London.
- Sweller, J., Merriënboer, J., & Paas, F. (2019). Cognitive architecture and instructional design: 20 years later. *Educational Psychology Review*, 31, pp. 261–292.
- Swidan, A. & Hermans, F. (2019). The Effect of Reading Code Aloud on Comprehension: An Empirical Study with School Students. *CompEd '19 Proceedings of the ACM Conference on Global Computing Education*. pp. 178–184.
- Szlávi, P., Zsakó, L. & Törley, G. (2019). Programming theorems have the same origin. *Central-European Journal of New Technologies in Research, Education and Practice*, 2019, Vol. 1, pp. 1–12.
- Szabó, Zs. (2017). Algoritmizálás tanítása Flowgorithm-mel. *InfoDidact'17. 10. Informatika Szakmódszertani Konferencia. Webdidaktika Alapítvány. Letöltve 2020.12.10.*  
<https://people.inf.elte.hu/szlavi/InfoDidact17/Manuscripts/SzZs.pdf>
- Székely, E. (1979). *Didaktika műszaki pedagógusok számára*. Tankönyvkiadó. Budapest.
- Teo, T.S.H. & Tan, M. (1999). Spreadsheet development and ‘what-if’ analysis: quantitative versus qualitative errors. *Accounting, Management and Information Technology*, 9, pp. 141–160.
- Tort, F. (2010). Teaching spreadsheets: Curriculum design principles. *Proceedings of EuSpRIG 2010 Conference*, pp. 99–110.
- Tort, F., Blondel, F-M. & Bruillard, É. (2008). Spreadsheet Knowledge and Skills of French Secondary School Students. *ISSEP 2008, LNCS 5090*, pp. 305–316.
- Tóth, P. (2015). *Szakmódszertan - polgári és biztonságvédelmi szakirány*. Typotop Kiadó, Budapest. Letöltve 2020.08.22. [https://regi.tankonyvtar.hu/hu/tartalom/tamop412b2/2013-0002\\_szakmodszertan\\_had-es\\_biztonsagvedelem/tananyag/JEGYZET-33-4.2.\\_A\\_rendszerezes\\_es\\_rogzit.html](https://regi.tankonyvtar.hu/hu/tartalom/tamop412b2/2013-0002_szakmodszertan_had-es_biztonsagvedelem/tananyag/JEGYZET-33-4.2._A_rendszerezes_es_rogzit.html)
- Varga, P., Jeneiné Horváth, K., Reményi, Z., Farkas, Cs., Takács, I., Siegler, G. & Abonyi-Tóth, A. (2020). *Digitális kultúra tankönyv 9*. Oktatási Hivatal. Budapest.
- Virágölglyi, P. (2004). *A tipográfia mestersége számítógéppel*. Osiris Kiadó Kft. Budapest.
- Vinacke, W. E. (1952). *The psychology of thinking*. McGraw-Hill, New York.

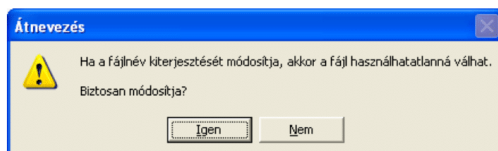
- W2 (2012). Report of JPMorgan Chase & Co. Management Task Force. Regarding (2012): CIO Losses. Letöltve 2020.08.10. [http://files.shareholder.com/downloads/ONE/2272984969x0x628656/4cb574a0-0bf5-4728-9582-625e4519b5ab/Task\\_Force\\_Report.pdf](http://files.shareholder.com/downloads/ONE/2272984969x0x628656/4cb574a0-0bf5-4728-9582-625e4519b5ab/Task_Force_Report.pdf)
- Wachsmuth, I. (1981). Two modes of thinking - also relevant for the learning of mathematics?. For the learning of mathematics 2(2). pp. 38–45.
- Wallas, G. (1926). The art of thought. Jonathan Cape, London.
- Webdidaktika Alapítvány (2017). InfoDidact'2017: 10. Informatika Szakmódszertani Konferencia: előadaskötet. Letöltve: 2021.02.10. <https://people.inf.elte.hu/szlavi/InfoDidact17/betolt.html>
- Webdidaktika Alapítvány (2018). InfoDidact'2018: 11. Informatika Szakmódszertani Konferencia: előadaskötet. Letöltve: 2021.02.10. <https://people.inf.elte.hu/szlavi/InfoDidact18/Infodidact2018.pdf>
- Webdidaktika Alapítvány (2019). InfoDidact'2019: 12. Informatika Szakmódszertani Konferencia: előadaskötet. Letöltve: 2021.02.10. <https://people.inf.elte.hu/szlavi/InfoDidact19/Infodidact2019.pdf>
- Wing, J. (2006). Computational thinking. Communications of the ACM, 49(3), pp. 33–35.
- Young, J. W. (1940). Technique for producing ideas. Advanced Publications, Chicago.

## 13. Melléklet

### 1. számú melléklet

#### Fájlkezelés teszt, a WDC módszer hatékonyságvizsgálatának elő- és utótesztje

F1. Mit jelent az alábbi üzenet adatfájlok esetén? (többet is megjelölhetsz)



- Ha a kiterjesztés változik a gép képtelen lesz felismerni a fájlt.
- Kiterjesztés módosításnál adatok elveszhetnek.
- Megváltozik a társítás és a fájl használhatatlan lesz.
- Megváltozik a társítás, de a fájl továbbra is használható lesz.
- Ha rossz formátumra írjuk át a fájlt, akkor az megsérül és használhatatlan lesz.
- Az új kiterjesztés más kódolása miatt a tartalom a felhasználó által olvashatatlan lesz.

F2. Mi történik, ha duplán kattintunk egy dokumentum fájlra (például: zz.jpg, zz.html, zz.ods, zz.xls)?

.....

.....

F3. Hogyan tudnál egy táblázatkezelő dokumentumot (.xlsx vagy .ods ) szövegfájlra (.csv vagy .txt) alakítani? (egyet jelölhetsz meg)

- Konvertálás.
- Exportálás.
- Átírjuk a kiterjesztést.
- Google-ben rákeresek.
- Mentés másként, kiválasztjuk az új típust.
- Importálás.
- Társítás.
- Mentés másként, átírjuk a kiterjesztést.
- Online konverter.
- Megnyitom Jegyzetkönyvben.

F4. Mi történik egy fájl kivágásakor? (többet is megjelölhetsz)

- Lomtárba kerül.
- Véglegesen törlődik.
- Megjelenik egy másik mappában.
- Vágólapra kerül.
- Nem történik semmi.
- Másolat készül róla.

F5. Válaszd ki, hogy milyen típusúak az alábbi fájlok! (többet is megjelölhetsz)

	szövegfájl	word-dokumentum	excel-munkafüzet	egyik sem
A.xls.txt	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
B.txt.xlsx	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
C.csv	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
D.docx.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
xlsx	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
...docx	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

F6. Döntsd el az alábbi állításokról, hogy igaz vagy hamis! Karikázd be a megfelelő betűjelet! (igaz = I, hamis = H, nem tudom = N)

1. Fájlok társítása a kiterjesztés alapján történik.	I	H	N
2. A társítás örök érvényű.	I	H	N
3. Mindig az a program van hozzátársítva az adatfájlhoz, amelyben készült.	I	H	N
4. Egy weblap megnyitható egyszerre szöveg- és egy táblázatkezelő programban is.	I	H	N
5. Egy weblap csak társítással nyitható meg szövegszerkesztőben.	I	H	N
6. Egy weblap linkkel (Ctrl+kattintással) nyitható meg egy szövegszerkesztőben.	I	H	N
7. A dokumentumfájlok társított fájlok.	I	H	N
8. Egy adatfájl kiterjesztésének átírásával a fájl típusa megváltozik.	I	H	N
9. Ha megnyitok egy szövegfájlt táblázatkezelőben, akkor automatikusan megváltozik a fájl típusa táblázatkezelői formátumra.	I	H	N
10. Ha megnyitok egy weboldalt szövegszerkesztőben, akkor automatikusan megváltozik a fájl kiterjesztése.	I	H	N
11. Egy weblap mentésekor átkonvertálható a fájl word-dokumentum típusra.	I	H	N
12. Szövegszerkesztő programban a tagolt szöveg táblázattá alakítható.	I	H	N

## 2. számú melléklet

### Szövegkezelés teszt, az ERM módszer hatékonyságvizsgálatának előtesztje

előteszt nyomtatott verzió	előteszt digitális verzió
<p><b>Gyógyszerek:</b></p> <p>Az orvosi szoba csak alkalmanként van nyitva. Legyen lehetőség <i>Lazmérió és fájdalomcsillapító ( a legkevésbé allergiás reakciót kelő )</i> gyógyszerek kérésére. Például az aszisztensnőől, vagy a titkárnőől.</p>	<p><b>Gyógyszerek:</b></p> <p>.....Az orvosi szoba csak alkalmanként van nyitva. Legyen lehetőség .....<i>Lazmérió és fájdalomcsillapító (a legkevésbé allergiás reakciót .....kelő.)</i> gyógyszerek kérésére. Például az aszisztensnőől, vagy a .....titkárnőől.</p>
<p><b>Tornaterem:</b></p> <p>Kévs óránk van a tornateremben! Szeretnénk, ha készítenének egy egyenletesebb beosztást.</p>	<p><b>Tornaterem:</b></p> <p>.....Kévs óránk van a tornateremben! Szeretnénk, ha készítenének egy .....egyenletesebb beosztást.</p>
<p><b>Toalettek:</b></p> <p>Kérnénk, hogy minden illemhelyen tegyék lehetővé a toalettpapír és a szappan használatát. Ha netán nincsen rá keret, kérhetnék a szil- lői munkaközösség segítségét.</p>	<p><b>Toalettek:</b></p> <p>.....Kérnénk, hogy minden illemhelyen tegyék lehetővé a toalettpapír és .....a szappan használatát. Ha netán nincsen rá keret, kérhetnék a szil- lői munkaközösség segítségét.</p>
<p style="text-align: right;">Tisztelettel: a 6.b osztály</p> <p>Debrecen, 2001. Február 13.</p>	<p style="text-align: right;">Tisztelettel: a 6.b osztály</p> <p>Debrecen, 2001. Február 13.</p>

### 3. számú melléklet

## Szövegkezelés teszt, az ERM módszer hatékonyságvizsgálatának utótesztje

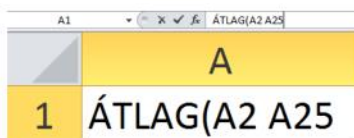
<p style="text-align: center;"><b>utóteszt nyomtatott verzió</b></p> <p style="text-align: center;"><b>KOCSITOLÓ VERSENY SZABÁLYZATA</b> <b>KOCS</b></p> <p><b>I. Résztvevők köre:</b></p> <p>Versegyen előzetes nevezés alapján, kornaiár nélkül, <u>születélekorára</u> bátki részt vehet. A férfi futamban vegyes csapattal is lehet nevezni. <u>Ruhái csak 15 év feletti fili-futár. A 18 év alatti kiskorúak nevezéséhez szülői írásbeli hozzájárulás szükséges, amekkor mind a nevezéskor mellettebbi szíveskedjenek.</u></p> <p><b>2. A versenyzőkre vonatkozó feltételek:</b></p> <p>A verseny jellegéből adódóan <u>AMATOR</u>-de fizikailag jól felkészült, egészséges versenyzőket várunk. Igazolati sportoló lehet, de <u>életműváltásértelen sportolók (PROFIN) a versenyen nem vehetnek részt.</u></p> <p>Kocsik kipróbálására, verseny helyszínének megismerésére a futamokat megelőző 2 nappal délutánokent lehetőséget biztosítanak a szervezők. Egyeztetni a Faluház telefonszáman lehet( 34/471-298 )</p> <p><b>3. A verseny nevezési díja:</b></p> <p>5 000.- Ft /csapat Csékket postai úton küldjük a jelentkezőknek, amelyet az Önkormányzat címére szíveskedjenek eljuttatni.</p> <p><b>4. A versenyre való jelentkezés módja és határideje:</b></p> <p><b>Móddja:</b> Levélben szíveskedjenek az Önkormányzat címére eljuttatni a nevezési lapot.</p> <p><b>Határideje:</b> A nevezéseket postai úton 2898 Kocs Faluház Komáromi út 5. címre, várjuk a versenyt megelőző 15 munkanappal június 20-ig. A szükséges szülői hozzájárulást is ezzel egy időben szíveskedjenek megküldeni. Kérjük ügyeljenek a nyomtatvány pontos kitöltésére, és a nevezési határidőre!</p> <p><b>5. Verseny leírása és helyszíne:</b></p> <p>Kocs községben a Tatai út START helytől a Faluház előtti CÉLIG, érintve a Faluközpontot. Távoltság kb. 1800 m.</p> <p style="text-align: right;"><b>Kocs Községi Önkormányzat</b></p>	<p style="text-align: center;"><b>utóteszt digitális verzió</b></p> <p style="text-align: center;"><b>KOCSITOLÓ VERSENY SZABÁLYZATA</b> <b>KOCS</b></p> <p><b>I. Résztvevők köre:</b></p> <p>Versegyen előzetes nevezés alapján, kornaiár nélkül, <u>születélekorára</u> bátki részt vehet. A férfi futamban vegyes csapattal is lehet nevezni. <u>Ruhái csak 15 év feletti fili-futár. A 18 év alatti kiskorúak nevezéséhez szülői írásbeli hozzájárulás szükséges, amekkor mind a nevezéskor mellettebbi szíveskedjenek.</u></p> <p><b>2. A versenyzőkre vonatkozó feltételek:</b></p> <p>A verseny jellegéből adódóan <u>AMATOR</u>-de fizikailag jól felkészült, egészséges versenyzőket várunk. Igazolati sportoló lehet, de <u>életműváltásértelen sportolók (PROFIN) a versenyen nem vehetnek részt.</u></p> <p>Kocsik kipróbálására, verseny helyszínének megismerésére a futamokat megelőző 2 nappal délutánokent lehetőséget biztosítanak a szervezők. Egyeztetni a Faluház telefonszáman lehet( 34/471-298 )</p> <p><b>3. A verseny nevezési díja:</b></p> <p>5 000.- Ft /csapat Csékket postai úton küldjük a jelentkezőknek, amelyet az Önkormányzat címére szíveskedjenek eljuttatni.</p> <p><b>4. A versenyre való jelentkezés módja és határideje:</b></p> <p><b>Móddja:</b> Levélben szíveskedjenek az Önkormányzat címére eljuttatni a nevezési lapot.</p> <p><b>Határideje:</b> A nevezéseket postai úton 2898 Kocs Faluház Komáromi út 5. címre, várjuk a versenyt megelőző 15 munkanappal június 20-ig. A szükséges szülői hozzájárulást is ezzel egy időben szíveskedjenek megküldeni. Kérjük ügyeljenek a nyomtatvány pontos kitöltésére, és a nevezési határidőre!</p> <p><b>5. Verseny leírása és helyszíne:</b></p> <p>Kocs községben a Tatai út START helytől a Faluház előtti CÉLIG, érintve a Faluközpontot. Távoltság kb. 1800 m.</p> <p style="text-align: right;"><b>Kocs Községi Önkormányzat</b></p>

#### 4. számú melléklet

### Táblázatkezelés teszt, az Sprego módszer hatékonyságvizsgálatának elő és utótesztje

Név: ..... Osztály: ..... Életkor.....  
Dátum:.....

- Hogyan egészítenéd ki az A1 cella tartalmát, hogy helyes képleted legyen? Írd a megoldásodat a pontozott vonalra!



- Kitaláltunk egy SPREGO() nevű függvényt. A függvénynek van két bemenő értéke: az első egy szöveg, a második egy egész szám. Karikázd be a helyesen írt képleteket!

- SPREGO("AAAA";2)
- SPREGO(aaaa;"2")
- =SPREGO("AAAA";2)
- =SPREGO(2;aaaa)
- =SPREGO(B1;2)
- =SPREGO(B1;C1+2)
- =SPREGO("AAAA";2;5)
- =SPREGO("B";3\*2)

- Hogyan hivatkozol az alábbi két cellatartományra?

	A	B	C	D	E
1	Country	Continent	Capital	Area	Population (thousand)
2	Afghanistan	Asia	Kabul	647500	27756
3	Albania	Europe	Tirana	28748	3545
4	Algeria	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

Hivatkozás a világosszürke tartományra: .....

Hivatkozás a sötétszürke tartományra: .....

4. Írd le, hogy szerinted a táblázatkezelő milyen sorrendben hajtja végre az alábbi képletben szereplő műveleteket! Az 1. lépés mellé azt a műveletet írd, amelyet a táblázatkezelő először hajt végre! (A táblázat sorainak száma nem szükségszerűen egyezik meg a lépések számával.)

=INDEX(A2:A5;SPREGO(B1;MAX(D2:D5)-7))

1. lépés	
2. lépés	
3. lépés	
4. lépés	
5. lépés	
6. lépés	

5. Oldd meg az alábbi feladatokat táblázatkezelő képletekkel!

	A	B	C
1	Név - kód	Ár	
2	Áfonya Hercegnő cukormentes tortája - KT057	3950 Ft	XXXX
3	Őrség zöld aranya torta ország tortája 2016 - KT058	3950 Ft	
4	Az ország tortája 2015 - KT050	3950 Ft	
5	Oreo keksz torta - KT052	3950 Ft	
56	Málnácska torta - KT056	3800 Ft	
57	Lekváros kardinális torta - KT044	3500 Ft	
58	Karamellás csoki mousse torta - KT098	3800 Ft	

- a. Írnod ki az összes tortának a kódját!  
 .....
- b. Írnod ki az összes tortának a nevét!  
 .....
- c. Az árak nem valódi számok, mert számjegyeket és betűket is tartalmaznak. Alakítsd át az összes árat valódi számmá!  
 .....
- d. A C2 cellába begépetünk egy árat. Írnod ki az összes tortánál, hogy **drága**, ha az ára nagyobb a C2-ben lévő árnál, minden más esetben írnod ki azt, hogy **olcsó**!  
 .....

## 5. számú melléklet

Név: ..... osztály: ..... dátum: .....

Oldja meg az a)–e) feladatokat táblázatkezelő képletekkel (MS Excel, OpenOffice, LibreOffice Calc), és mondja meg, hogy mit csinál az f) képlet!

	A	B	C	D	E	F	G
1	ország	földrész	főváros	terület	lakosság (ezer)		
2	Afghanistan	Asia	Kabul	647500	27756		
3	Albania	Europe	Tirana	28748	3545		
4	Algeria	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		
233	Yemen	Asia	Sanaa	527970	18701		
234	Yugoslavia	Europe	Belgrade	102350	10657		
235	Zambia	Africa	Lusaka	752614	9959		
236	Zimbabwe	Africa	Harare	390580	11377		

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 G2!

.....

e) Írassa ki azon országok számát, amelyek területe nagyobb, mint G2!

.....

f) Mit csinál az alábbi összetett képlet?

`{=SZUM(HA(B2:B236="Europe";HA(BAL(A2:A236)="A";1)))}`

.....

## 14. Publikációs jegyzék

### Idegen nyelvű tudományos közlemények nemzetközi folyóiratban

Gábor Csapó, Katalin Sebestyén, Mária Csernoch, Kálmán Abari (2021). Case study: developing long-term knowledge with Sprego. *Education and Information Technologies* 26, pp. 965-982.

### Idegen nyelvű tudományos nemzetközi konferencia közlemények

Katalin Sebestyén, Gábor Csapó, Mária Csernoch (2019). Introduction to Algorithmic Based Data Management in Spreadsheet Environment. *The Turkish Online Journal of Educational Technology*. INTE 2019, pp. 354-361. Retrieved 08/06/2020 from [https://www.int-e.net/publication\\_folder/inte/inte\\_iticam\\_2019.pdf](https://www.int-e.net/publication_folder/inte/inte_iticam_2019.pdf)

Piroska Biró, Mária Csernoch, Katalin Sebestyén, Eszter Gombos (2018). Algorithmic Skills Vs. Time Spent On Computers And Mobile Devices. *Turkish Online Journal of Educational Technology*, INTE 2018 November, pp. 133-140. Retrieved 05/05/2020 from [http://www.tojet.net/special/2018\\_12\\_2.pdf](http://www.tojet.net/special/2018_12_2.pdf)

Katalin Sebestyén, Gábor Csapó (2018). Visualising Sprego Inequality Problems With 2d Representations. *Turkish Online Journal of Educational Technology*, INTE 2018 November, pp. 1081-1091. Retrieved 05/05/2020 from [https://www.int-e.net/publication\\_folder/inte/inte\\_iticam\\_idec2018\\_v2.pdf](https://www.int-e.net/publication_folder/inte/inte_iticam_idec2018_v2.pdf)

Gábor Csapó, Katalin Sebestyén (2017). Educational Software for the Sprego Method. *The Turkish Online Journal of Educational Technology*, 2017 October, pp. 987-1000. Retrieved 06/06/2020 from [http://www.tojet.net/special/2017\\_10\\_1.pdf](http://www.tojet.net/special/2017_10_1.pdf)

### Idegen nyelvű tudományos hazai konferencia közlemények

Katalin Sebestyén (2020). Students' Knowledge in File Management. *Proceedings of the 11th International Conference on Applied Informatics (ICAI 2020)*

### Magyar nyelvű tudományos konferencia közlemények

Csapó Gábor, Sebestyén Katalin (2015). Oktatóprogram a Sprego táblázatkezelő módszerhez. *INFODIDACT 2015: Informatika Szakmódszertani Konferencia* (ed. Szlávi Péter, Zsakó László). <http://people.inf.elte.hu/szlavi/InfoDidact15/Manuscripts/CsGSK.pdf> Letöltés dátuma: 2016.01.01. ISBN 978-963-12-3892-1

Sebestyén Katalin (2014). Te is lehetsz informatikatanár (?). *INFODIDACT 2014: Informatika Szakmódszertani Konferencia* (ed. Szlávi Péter, Zsakó László). <http://people.inf.elte.hu/szlavi/InfoDidact14/Manuscripts/SK.pdf> Letöltés dátuma: 2016.01.01. ISBN 978-963-12-0627-2

## **Beküldött, közlésre elfogadott tudományos közlemények**

Katalin Sebestyén, Gábor Csapó, Mária Csernoch, Bernadett Aradi. Error Recognition Model: End-user Text Management. Acta Polytechnica Hungarica

Katalin Sebestyén, Gábor Csapó, Mária Csernoch, Kálmán Abari. Measuring the algorithmic skills of students working with low- and high-mathability programming approaches. Acta Polytechnica Hungarica

## **Szakmai konferenciák, előadástartások idegen nyelven**

<b>Időpont</b>	<b>Előadás adatai</b>
2020. január	Students' Knowledge in File Management After Elementary School, The 11th International Conference on Applied Informatics tudományos konferencia, Eger
2019. július	Introduction to algorithmic based data management in spreadsheet environment, International Conference on New Horizons in Education nemzetközi tudományos konferencia, Prága
2018. július	Visualising Sprego inequality problems with 2D representations, International Conference on New Horizons in Education nemzetközi tudományos konferencia, Párizs
2018. július	Algorithmic Skills Vs. Time Spent On Computers And Mobile Devices, International Conference on New Horizons in Education nemzetközi tudományos konferencia, Párizs
2017. július	Educational Software for the Sprego Method, International Conference on New Horizons in Education nemzetközi tudományos konferencia, Berlin

## **Szakmai konferenciák, előadástartások magyar nyelven**

<b>Időpont</b>	<b>Előadás adatai</b>
2019. május	Bevezetés az algoritmus alapú adatkezelésbe táblázatkezelői környezetben: Digitális Témahét tapasztalatai, ADA 2019 tudományos konferencia, Debrecen
2019. április	Digitális Témahét: Informatika didaktika módszertani bemutató szakórák, Szlovák Tanítási Nyelvű Óvoda, Általános Iskola, Gimnázium és Diákotthon, Budapest
2019. április	Digitális Témahét: Informatika didaktika módszertani bemutató szakórák, Bródy Imre Gimnázium, Ajka

2018. február A számítógépes gondolkodás fejlesztésének Sprego eszközei szimpóziumi előadás részeként: Számítógépes gondolkodás fejlesztése számítógép nélkül – a Sprego programozás unplug-ged eszközei, Oktatás-Informatika-Pedagógia Konferencia 2018, Debrecen
2017. március Sprego oktatószoftver, XXXIII. OTDK Konferencia Tanulás-és Tanításmódszertani – Tudástechnológiai Szekció, különdíj, Győr
2016. május Tanulási segédanyag a Sprego táblázatkezelői módszerhez, Debreceni Egyetem Hatvani István Szakkollégium Tavaszi Tudományos Hallgatói Konferenciája, Debrecen
2016. április Oktatóprogram a Sprego módszerhez, XVII. Eötvös Konferencia, Budapest
2015. november Oktatóprogram a Sprego táblázatkezelő módszerhez, InfoDidact Konferencia, Zamárdi
2015. november Sprego oktatóprogram, Debreceni Egyetem Informatikai Kar TDK, Debrecen, 1. hely
2015. november Tanárszakosoknak oktatószoftverek fejlesztése vizuális programozási eszközökkel, Debreceni Egyetem Informatikai Kar, Bevezetés az informatikába tanóra, Debrecen
2015. május Játék az informatikaórán, Játékkal a világ körül nemzetközi tudományos konferencia, Hajdúböszörmény
2015. május Informatikaoktatás a 21. század elején Magyarországon, Hatvani István Tavaszi Tudományos Hallgatói Konferencia, Debrecen
2015. április Miből lesz a cserebogár, az informatikatanár?, OTDK, Debrecen
2014. november Te is lehetsz informatikatanár(?), InfoDidact Konferencia, Zamárdi
2014. április Klikk és Klavia Túra - Kiből lehet informatikatanár?, XV.Eötvös Konferencia, Budapest
2014. április Miből lesz a cserebogár, az informatikatanár?, Debreceni Egyetem Informatikai Kar TDK, Debrecen, 1. hely
2014. március Információs társadalom és a könyvtár, 10. Informatikai Szakmai Napok, Debrecen
2013. október Miért nem tudok többet informatikából, mint egy bölcsész?, 9. Informatikai Szakmai Napok, Debrecen

2013. április      Klikk és Klavia Túra- Informatika tankönyvek terminológiai következtetései, OTDK, Eger
2012. december    Informatika tankönyvek terminológiai vizsgálata, Hallgatói Konferencia, Debrecen
2012. október      Informatika tankönyvek terminológiai következtetései (társszerző), Törökbálint
2012. november    Informatika tankönyvek terminológiai következtetései, Debreceni Egyetem Informatikai Kar TDK, Debrecen, 1. hely
2012. május        Napjaink női irodalmának olvasásszociológiai vizsgálata, Debreceni Egyetem Informatikai Kar TDK, Debrecen
2012. május        Klikk és Klavia Túra - Terminológiai következtetések és azok hatása a leggyakoribb beviteli eszközök használata során, Debreceni Egyetem Informatikai Kar TDK, Debrecen, 2. hely