

Doktori (PhD) értekezés tézisei

**Séma- és tudástranzferalapú számítógépes
problémamegoldási képességek fejlesztése
az informatikai alkalmazói ismeretek taní-
tása során**

Sebestyén Katalin

Témavezető: Dr. Csernoch Mária



DEBRECENI EGYETEM
Informatikai Tudományok Doktori Iskola

Debrecen, 2021

Tartalomjegyzék

1. Bevezetés	1
2. Elméleti háttér.....	5
3. Fájlkonverzió a fájlkezelés témakör tanításához	6
3.1 Fájlkezelési ismeretek mérése.....	6
3.2 Webtábla→adattábla konverzió	8
4. Hibefelismerés- és keresés a szövegkezelés tanításához	10
4.1 Széles körben elterjedt szövegkezelési módszerek	10
4.2 Szövegkezelési ismeretek mérése	11
4.3 Hibafelismerési Modell.....	12
5. Sprego a táblázatkezelési ismeretek oktatásához.....	14
5.1 Széles körben elterjedt táblázatkezelői megközelítések.....	14
5.2 Sprego programozás és adatkezelés	15
6. Hipotézisek	15
7. Jövőbeli elképzelések	22
8. Irodalomjegyzék	24
1. Introduction	33
2. Theoretical background	38
3. Webtable→datatable conversion for the file management topic	39
3.1 Measuring file management skills.....	39
4. The Error Recognition Model (ERM) for the text-management topic.....	42
4.1 Widespread methods	42
5. Spreadsheet Lego (Sprego) for spreadsheet management	45
5.1 Widespread spreadsheet approaches	45
6. Hypotheses.....	47
7. Future plans	53
8. References	55

1. Bevezetés

Informatikai alkalmazói ismeretekre nem csak a diákoknak és az irodai munkavállalóknak van szükségük, hiszen a digitális világ beépült a mindennapjainkba, ez legyen digitális oktatás, továbbképzés, gyermekeink e-naplójának nyomon követése, kapcsolattartás vagy éppen hivatalos ügyintézés. Az informatikai alkalmazói ismeretek és a számítógépes gondolkodás mint alapképesség elsajátításának első irányított szintere a közoktatás, ahol fiatalon, tanári útmutatással kialakítható egy hatékony problémamegoldó szemléletmód. A digitális bennszülöttek (Prensky 2001; Jukes & Dosaj 2006, Szabados 2009; Jukes et al. 2010), azok akik már a születésüktől kezdve együtt nőttek fel a digitális eszközökkel, tehát anyanyelvi szinten értik és beszélik a digitális technológiák, okoseszközök, számítógépek, játékkonzolok és az internet nyelvét. Ezeknek az eszközöknek a használatát többnyire autodidakta módon, próbálgatások és tapasztalataik alapján sajátítják el a fiatalok. Ez azonban, hasonlóan az írás, olvasás, számolás alapképességekhez (Reading, wRiting, aRithmetics, 3R) (Wing 2006) nem garantálja azt is, hogy helyes az az ismeret, amit így önmaga kitapasztalt, és azt sem, hogy alkalmazható hatékony problémamegoldásra. A közoktatás egyik legfontosabb szerepe, hogy a hozott ismereteket pontosítsa, kiegészítse és rendszerezze. A tudományterület fejlődése szempontjából kiemelten fontos, hogy a tanulók megismerjék az informatikához kapcsolódó kifejezéseket és terminológiát, hogy a felmerülő problémákat, útmutatásokat természetes nyelven is meg tudják fogalmazni, meglássák és megértsék az eszközök felülete mögött rejlő összefüggéseket.

Nincs egyetértés abban, hogy a digitális bennszülötteknek szüksége van-e szervezett, iskolai keretek között folyó informatikaoktatásra, a tanulók többsége is feleslegesnek tartja. Az utóbbi években azonban több vizsgálat bizonyította, hogy a digitális generáció tagjának lenni, nem foglalja magába a magas szintű digitális műveltséget és a számítógépes gondolkodást mint alapképességet (Prensky 2001; Kirschner & Bruyckere 2017; U. S. Department of Education 2018). Bizonyításra került az is, hogy az okostelefonok mindennapi használata nem segíti a számítógépes és az algoritmikus gondolkodás fejlesztését, fejlődését (Csernoch & Biró 2019).

Elérkezett tehát az az időszak, amikor a számítógépes gondolkodás, a három alapképesség mellett megjelenhet negyedik alapképességként (Kirschner et al. 2006; Wing 2006). Ahogy a gyermek a szülőktől megtanul beszélni, ettől függetlenül szüksége van nyelvi képzésre, és ezt senki sem kérdőjelezi meg. Nincs ez másképp a számítógépes gondolkodás esetében sem.

Az informatikaoktatás, ezen belül is az alkalmazói ismeretek fontosságát és szükségességét alátámasztja az is, hogy a meglévő szöveges- és táblázatkezelői dokumentumok 80–90%-a hibás (Ben-Ari 1999; Panko 2008; Panko & Aurigemma 2010, Csernoch & Biró 2015a). Ezek a hibák komoly anyagi veszteségeket generálnak akár a szerzőknek, akár a megrendelőknek (Eusprig 2020).

A 2009-es PISA Student Online (OECD 2011) a tanulók digitális kompetenciáinak mérésére irányuló felmérés, ahol a résztvevő 19 ország között szerepelt Magyarország is. A magyar tanulók eredményei elmaradnak az OECD átlagtól, csupán három ország tanulóinál tudunk jobb eredményt elérni (1. ábra). Ebben a mérésben a tanulók digitális jártasságát 4+1 képességszint alapján határozták meg (1. táblázat). A belépő szintet (2. szint) el nem érő tanulók tudását az 1. táblázatban szereplő magyarázatok mellett úgy jellemezték, hogy „... nem tudnak információt keresni az interneten csak magukra hagyatkozva, azaz explicit útmutatás nélkül, és kihívást jelent számukra a források hitelességének és relevanciájának kritikus megítélése.” (Balázi & Ostorics 2011).

A magyar tanulók fele a 2. jártassági szintet (24,97%) vagy azt sem (26,82%) érte el, másszóval digitális analfabéták. A 2012-es mérésben ezeknek a tanulóknak a száma tovább emelkedett (OECD 2015). Az alacsony digitális kompetencia szint mellett arra is rámutatott a felmérés, hogy a legjobban és a legrosszabban teljesítő tanulók (5%) eredménye közötti különbség hazánkban a legnagyobb, ami az oktatási színvonal egyenetlenségére utal (OECD 2011).

képességszint	Mit kell tudniuk az adott szintet elérő tanulóknak?
5. szint és afölött	Szokatlan kontextusú, nem egyértelmű információ elhelyezése, elemzése és kritikus értékelése. A szöveg értékeléséhez szempontokat kell találni. Számos oldalon keresztül kell navigálni (egyértelmű utasítás nélkül), és különféle formátumú szövegeket kell megvizsgálni.
4. szint	Több forrásból származó, különféle oldalakról, különféle formátumú szövegekből összegyűjtött információk értékelése; kritériumok alkotása ismerős kontextusú információ értékelésére. Komplex jelentések megalkotása jól meghatározott tudományos vagy technikai kritériumok alapján.
3. szint	Információk integrálása: pontosan meghatározott célinformációhoz való eljutás több oldalon keresztül vagy egyszerű kategóriák alkotása. Közvetlenül hozzáférhető információ értékelése
2. szint	Pontosan meghatározott, általában ismerős témákhoz kötődő információ megkeresése és értelmezése. Egyszerre kevés oldalon keresztüli navigáció, webes eszközök (pl. legördülő menü) használata. Különféle formátumokban megjelenő információk integrálása, világosan körülhatárolható kategóriákba tartozó példák felismerése.
2. szint alatt (+)	

1. táblázat PISA Student On Line 2009-es felmérésben megfogalmazott képességszintek, ahol a 2. szint a belépő szint. Ezen szintek mellett egy újabb kategóriát is meghatároztak, ahová a belépő szintet el nem érő tanulók kerültek besorolásra. (OECD 2011; Balácsi & Ostorics, 2011)

Az elemzők összevetették a tanulók digitális kompetencia szintjét az iskolai számítógéphasználattal, melyek között a legnagyobb negatív korrelációt Magyarországon találták (1. ábra).

Table VI.A
AN OVERVIEW OF PERFORMANCE IN DIGITAL READING, NAVIGATION AND COMPUTER USE

	Digital reading performance		Gender difference in digital reading scores between boys and girls	Index of number of relevant pages visited (navigation skills)	Computer use at home			Computer use at school				
	Mean score	Score dif.			%	% dif.	Score dif.	%	% dif.	Score dif.	%	Score dif.
OECD average	499	-24	46.3	92.3	16.0	80	74.2	0.3	9			
Korea	568	-18	52.8	87.5	19.5	49	62.7	3.5	2.1			
New Zealand	537	-40	49.7	92.5	20.2	90	83.4	6.4	20			
Australia	537	-28	49.6	96.7	7.8	84	91.6	5.6	42			
Japan	519	-23	50.1	75.9	38.6	48	59.3	2.6	14			
Iceland	512	-30	47.5	99.1	1.2	74	79.5	5.1	22			
Sweden	510	-26	47.8	97.7	4.7	105	89.1	4.7	28			
Ireland	509	-31	47.4	93.2	10.9	60	62.9	0.4	-3			
Belgium	507	-24	47.7	96.9	9	102	62.8	-1.1	9			
Norway	500	-35	46.9	98.7	2.7	77	93.0	2.5	25			
France	494	-20	46.1	m	m	m	m	m	m			
Denmark	489	-6	47.2	98.8	2.8	79	93.0	1.8	6			
Spain	475	-19	44.2	92.6	14.4	78	65.5	-4.0	11			
Hungary	468	-21	41.6	91.8	23.6	102	69.3	-8.9	-27			
Poland	464	-29	42.0	92.1	22.9	84	60.6	-9.1	8			
Austria	459	-22	43.3	98.2	3.7	94	84.1	-3.2	-6			
Chile	435	-19	37.7	73.2	60.3	69	56.8	-2.0	2			

1. ábra A PISA Student On Line 2009 digitális kompetencia-mérés eredménye, kiemelve a magyar tanulók iskolai számítógéphasználata és a felmérésben elért eredmények közötti korreláció.

Ezen az eredményen az oktatási rendszerünk a következő Nemzeti alaptantervre épülő (OFI 2012a), 2013-ban bevezetett kerettantervekben (OFI 2012b) megfogalmazott drasztikus óraszám csökkenéssel (15) kívánt változtatni (2. táblázat). Ha nincs magas óraszám, akkor nincs negatív korreláció sem.

	évfolyam											
	1	2	3	4	5	6	7	8	9	10	11	12
2009			0,5	0,5	0,5	1	1	1	1,5	1	1,5	1,5
2012						1	1	1	1	1		
2020			1	1	1	1	1	1	2	1	2	

2. táblázat A 2009-es, a 2012-es és a 2020-es kerettantervekben javasolt heti informatikaóraszám évfolyamokra bontva (OFI 2008, 2012, 2020).

A 2013-ban bevezetett kerettanterv (OFI 2012b) számos kritikát kapott, ami elsősorban az óraszámcsökkentésből és a változatlan tananyagtartalom-ból fakadt. A pedagógusok teljesíthetetlennek találták a tananyag mennyiségét a rendelkezésre álló órakeretben. Az informatikatanárok feladatát tovább nehezítette, hogy a kerettantervben az elsajátítandó tananyag megfogalmazása túl általános és semmitmondó (Nagy & Csernoch 2018):

- **egyszerű** felhasználói szoftverek alapszintű kezelése,
- **rövid** rajzos dokumentum létrehozása,
- az operációs rendszer **alapvető** funkcióinak ismerete,
- **minta** alapján szerkesztés.

Ez a szövegezés lehetőséget adott arra, hogy a pedagógusok sajátosan értelmezzék a követelményeket, attól függően, hogy mennyire mély a tudásuk az adott területen.

A 2020/2021-es tanév kezdetén bevezetésre került egy újabb NAT (OFI 2020), amelyekben megemelkedett az informatikaórák száma (↑6) a 2012-es kerettantervhez (OFI 2012b) képest (2. táblázat), valamint a témakörök is át-szervezésre kerültek. Bővült a programozáshoz kapcsolódó témakörök száma, a könyvtárinformatika pedig teljes egészében kikerült az informatika tantárgyból. Továbbra is igaz, azonban, hogy a megnevezett témakörök tanítására nem elegendő a tantárgyhoz rendelt óraszám.

2. Elméleti háttér

Az informatikaoktatásnak szemléletváltásra van szüksége, a gyakorlatban leginkább elterjedt és alkalmazott, ugyanakkor rendkívül alacsony hatékonyságú felületi megközelítést el kell hagyni (Ben-Ari 1999; Panko 2008; Eusprig 2020). Jelenleg elterjedt módszer, hogy a témaköröket egymástól elszeparálva a szoftverkörnyezetre fókuszálva oktatják. A szövegezés kifejezés alatt a Word szövegszerkesztői programot, és annak a felületén történő navigációt értik a tanárok és a tanulók, mellőzve a dokumentumtervezést, a hiteles források keresését és megjelölését, a jólformázott szöveg ismérveit, a szöveg tördelési, formázási, szintaktikai, szemantikai és tipográfiai szabályait. Az informatikaóráknak nem egy szoftverkörnyezet megismeréséről kellene szólnia, hanem a digitális kompetenciák, digitális írástudás, a számítógépes gondolkodás fejlesztéséről és a problémamegoldási stratégiák elsajátításáról,

amelyek egy univerzális és hosszútávú tudás elérését biztosítják. A változást nem csak a PISA eredmények indokolják, hanem a jövő kihívásai és a tanulók indokolatlanul magas informatikai önbizalma és önértékelése.

A kutatásaim során a valós életből származó problémák megoldásán alapuló módszerek fejlesztését, alkalmazását és hatékonyságvizsgálatát végeztük el a kutatócsoport tagjaival. A módszerek alapja a Pólya György által kidolgozott koncepcióalapú (Pólya, 1954) problémamegoldási megközelítés, amely más tudományterületeken és az informatikán belül a programozásoktatásban bizonyítottan hatékony és elterjedt megközelítés (Hom et al. 2020).

Kutatásaink során, nagy hangsúlyt fektettünk az informatikán belüli tudástranszferelemek aktivizálására. Ezen keretek között az informatikai témakörhöz kidolgozott módszerek és eszközök, informatikai tartalmak kapcsolódnak egymáshoz, valamint szervesen bekapcsoltuk a számítógépes problémamegoldási stratégiánkba a más tudományterületekről érkező tartalmakat. A megközelítés erősen hagyatkozik a korábban elsajátított ismeretekre, a sémákban való gondolkodásra, amely kulcsszerepet játszik a hosszútávú tudás kialakításában (Sweller 1988, Sweller et al. 2011, Kahneman 2011).

3. Fájlkonverzió a fájlkezelés témakör tanításához

3.1 Fájlkezelési ismeretek mérése

A kerettantervekhez igazodva, legelső témakörök között szerepel a fájlkezelés, az operációs rendszer használatának elsajátítása. Visszajelzések és tapasztalatok alapján ezt a témakört gyakran elhanyagolják a tanárok a „digitális bennszülöttek” feltételezett hozott tudása miatt. A témakör indokolatlan ignorálását és fontosságának alábecslését igazolja a 2018 júniusában végzett országos mini kompetencia teszt, amely keretek között a tanulók számítógép és internethasználati szokásai mellett informatikakompetencia mérésre került sor (Nagy & Csernoch 2018). Kutatásaim egyik fő iránya a középiskolát megkezdő tanulók általános iskolából hozott informatikai ismereteinek feltérképezése. Ennek megfelelően, a mini kompetencia teszt 8880 résztvevője közül a 7. (1562 fő) és a 8. évfolyamos (1643 fő) tanulók válaszainak elemzését végeztem. A vizsgálatban a tanulók, önértékelés alapján megjelölték a jártassági szintjüket különböző informatikai témakörökben, valamint jelölhették

azt, hogy a témakört iskolában sajátította-e el (2. ábra). A fájlkezelés témakör esetében a tanulók 79,06% választotta a magasabb, 3–5 jártassági szinteket. A tanulók 14,72%-a vallotta azt, hogy nem tanult fájlkezelést iskolában. Azok a tanulók, akik nem tanultak fájlkezelést, ugyanolyan magabiztosak a tudásukban, mint a többiek (nem tanult 4. szint 16,40%; 5. szint 24,04%) Ez az eredmény látszólag igazolja, hogy a „digitális bennszülötteknek” nincs szükségük a fájlkezelés oktatására.

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

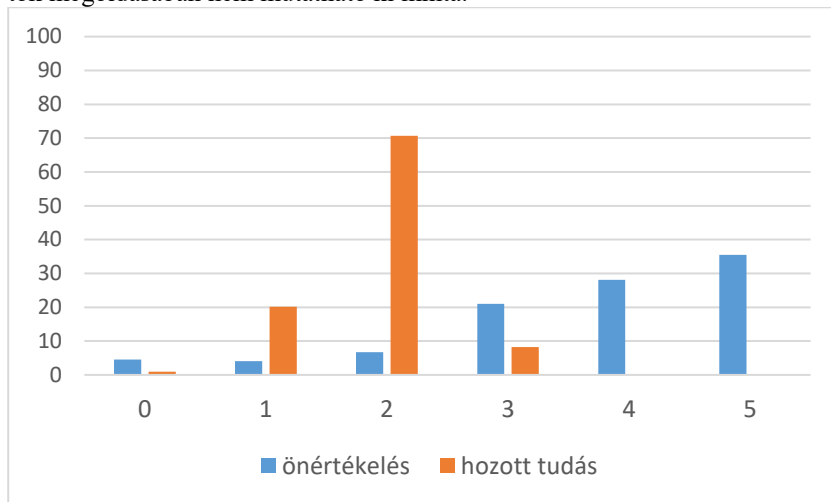
	Mennyire ismered a témakört?						Iskolában tanultad?	
	Karikázd be a megfelelő számot!						tanultam	nem tanultam
fájlkezelés	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
szövegkezelés	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
táblázatkezelés	0	1	2	3	4	5	<input type="checkbox"/>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

2. ábra A mini kompetencia tesztben az informatikai témakörök jártassági szintjének meghatározása önértékelés alapján (Nagy 2018).

A tanulók önértékelésének eredményét összehasonlítottuk a fájlkonverzió (webtábla→adattábla konverzió, webtable→datatable conversion, WDC) módszer hatékonyságvizsgálata során végzett előteszt eredményeivel, amely a középiskolát megkezdő tanulók fájlkezelési ismereteit méri a témakör oktatása előtt. A felmérésben két középiskola 9. osztályos tanulói vettek részt, összesen 109 fő. Az összehasonlítás érdekében a tanulói eredményeket megfeleltettük a 0–5-ig terjedő skálának. Az így kapott eredmény alapján egyetlen tanuló jártassági szintje sem éri el a 4–5. szintet (3. ábra) (Sebestyén 2020). A tanulók önértékelése és valós tudása közötti eltérést magyarázza a Dunning-Kruger effektus, mely szerint minél kevésbé jártas valaki egy területen, annál inkább érzi biztosnak a tudását (Kruger & Dunning 1999), mivel hiányzik az a tudás, amely szükséges ahhoz, hogy reálisan felmérje tudását. Röviden, azt se tudja, hogy mit nem tud.

A vizsgálatok összehasonlítása rámutatott arra, hogy a „digitális bennszülötteknek” is szükségük van az informatikaoktatására, még egy olyan alapvető ismereteket megkövetelő, látszólag egyszerű témakör esetében is, mint

a fájlkezelés. A tanulók hozott ismereteinek vizsgálata során azt tapasztaltuk, hogy a tanulók tudása rendszerezetlen, az azonos tudáselemet igénylő feladatok megoldásában nem mutatható ki minta.



3. ábra A mini kompetencia teszt eredménye (önértékelés) és a WDC módszer hatékonyságvizsgálatának előteszt eredménye (hozott tudás).

3.2 Webtábla→adattábla konverzió

A témakör oktatására a kutatócsoportunk által részletesen kidolgozott fájlkonverziós módszert alkalmazzuk. A módszer lényege, hogy a tanulási-tanítási folyamat során a tanulók életkori sajátosságainak és érdeklődési körének megfelelő, interneten elérhető autentikus tartalmak adatfeldolgozását végezzük fájlkonverziók és egyéb adatmanipulációs műveletek sorozatán keresztül (Csernoch & Dani 2017, Csernoch & Biró 2019). A módszer ötlete szorosan kapcsolódik a Sprego táblázatkezelői módszer oktatásához (Csernoch 2014), ahol a tanulók – az iskolai gyakorlattól eltérően – nem tartalom nélküli üres táblázatokkal, illetve fiktív, értelmetlen adatokkal dolgoznak (Papp & Csernoch 2019), hanem valós tartalmakkal és forrásokkal. Az interneten található táblázatok és táblázatoknak tűnő szerkezetek (webtábla) sok esetben nem emelhetők át egy az egyben táblázat- vagy adatbáziskezelő környezetbe annak érdekében, hogy feladatokat, számításokat lehessen végezni rajtuk.

A WDC módszer során a tanulók elsajátítják, hogy hogyan lehet ezeket a forrásokat a konverziós lépések sorozatán keresztül egy normalizált táblává alakítani. A WDC problémamegoldás során a tanulók adatelemzési, algoritmizálási, majd ezt követően fájlkezelési, valamint szöveg- és táblázatkezelőben elvégezhető adatmanipulációs műveleteket hajtanak végre. A konverzió során a tanulók számítógépes gondolkodására helyezük a hangsúlyt, amely elősegíti a tudatos eszközválasztást is. A tanulók megtanulják az operációs rendszerhez, fájlműveletekhez tartozó szakkifejezéseket, amelyek segítségével precízen meg tudják fogalmazni a megoldás algoritmusát és műveleteit. A tanulók a feladatmegoldás során több fájl típust használnak, ellenpéldákat mutatunk arra, hogy egy adatfájl csak a hozzá társított programmal nyitható meg, valamint arra, hogy a kiterjesztés megváltozása során a fájl használhatatlanná válik és megsérül. A konverzió a tanulóktól kreatív megoldásokat, ötleteket vár, amelyeket a tanár a Pólyától származó kutatás alapú tanulás (directed inquiry-based) (Pólya 1954) vagy ehhez a megközelítéshez szorosan köthető, napjainkban egyre népszerűbb coaching technikát (Chmielewska & Gilányi 2019) alkalmazva tudja irányítani a problémamegoldást. Kiemelten fontosnak tartjuk, hogy a tartalom nélküli, szövegkörnyezetből kiragadott tankönyvi és ECDL feladatokkal ellentétben (Papp & Csernoch 2018; Dancsó & Korom, 2013; ECDL Foundation 2019) a tanulók sémaépítési és -alkalmazási képességeit fejlesszük (Sweller 1988; Sweller et al 2011), 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 részt vegyenek a valós világ problémáinak megoldásában.

A problémamegoldás során alkalmazott lépések minden konverzió esetében változatosak, így kreatív megoldásokat is igényel a konverzió keretét alkotó séma alkalmazása mellett. A tanulók az ötleteiket minden esetben közösen, természetes nyelven fogalmazzák meg, amely nyitottabbá teszi a tanulókat, és lehetőséget ad a gondolkodásra, egymás ötleteinek mérlegelésére, átalakítására.

4. Hibefelismerés- és keresés a szövegkezelés tanításához

4.1 Széles körben elterjedt szövegkezelési módszerek

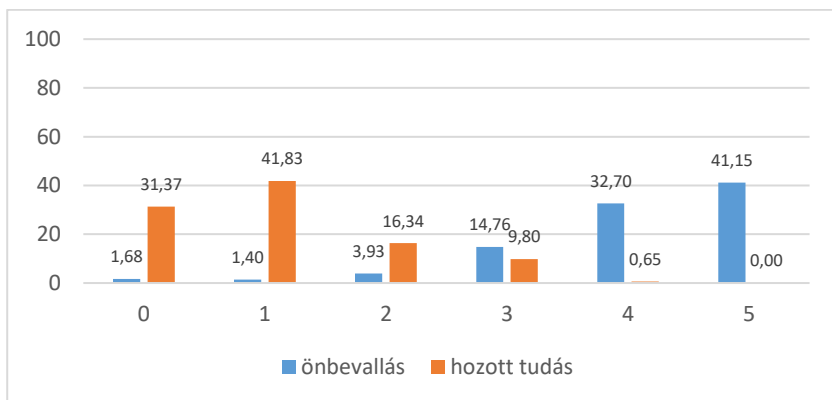
A közoktatásban az egyik leginkább hangsúlyos informatikai alkalmazói témakör és a gyakorlatban leginkább alkalmazott ismeret a szövegkezelés. Az alkalmazói ismeretek oktatására jellemző a szoftvercentrikusság, a kezelői felülethez tartozó ismeretek átadása. Nincs ez másképp a szövegszerkesztés és a Word szövegszerkesztő program esetében sem. A tankönyvekre jellemző a tartalom nélküli szövegek szerkesztése, formázása – nem ritkán hibás formázása –, melynek hagyományát a hivatalos tankönyvcsaládok tovább örökítik (Bánné et al. 2008; Bártfai 2011; Dancsó & Korom 2013; Lakosné et al. 2019a, 2019b, 2019c; Varga et al. 2020). Az alkalmazói ismerettel foglalkozó tankönyvek, szinte kizárólagosan a programablak részeivel és felületének ismertetésével kezdődnek. Ezeknek az ismereteknek az ismételt oktatása minden egyes szoftverkörnyezetben teljesen felesleges. Ennek magyarázata, hogy a programablakok azonos felépítésűek, így sokkal inkább az operációs rendszerek tanításához, mintsem az egyes alkalmazói szoftverekhez tartoznak. Másrészt nem kelti fel a tanulók figyelmét, unalmas. Nem felelkezhetünk meg továbbá arról sem, hogy nem a szoftverfelület tanítása a cél, hanem a hatékony problémamegoldás. Jellemző továbbá ezen felületek folyamatos változása, átalakítása. A tankönyvek illusztrációk a felületen történő navigációt, a menüszalag részeit, a különböző karakter- és bekezdésformázásokat szemléltetik, gyakran egyetlen kitalált szövegen keresztül. A felületalapú megközelítés azon túl, hogy csak az adott verziójú szoftver felületének megismerését teszi lehetővé – újabb verzió esetében kezdhetjük előlről a tanulási folyamatot –, az átadott ismeretek nem eredményeznek hosszútávú tudást, mivel nem köthető tartalomhoz.

A természetes nyelvű szövegkezelést nehezíti, hogy mai napig nem léteznek megbízható hibakeresők, automatizált ellenőrzők, ami a mesterséges informatikai nyelvek esetében adott. Ugyan léteznek helyesírásellenőrző programok, de ezek hatékonysága függ a nyelvtől, a helyes tördeléstől, a szerző nyelvhasználati szokásaitól. A hatékonyságot tehát mindenképpen befolyásolja a felhasználó háttérismerete. A megbízható hibakeresők hiányában, a felhasználók elsősorban a nyomtatott dokumentum vagy a nyomtatási kép

megjelenését veszik alapul, ott ellenőrzik az elkészült szöveget, ha azt „megfelelőnek” ítélik, akkor elégedettek a munkájukkal, függetlenül attól, hogy a digitális dokumentum helyes-e.

4.2 Szövegkezelési ismeretek mérése

A fájlkezeléshez hasonlóan, a szövegkezelés esetében is megvizsgáltuk a mini kompetencia teszt önértékelés részében adott 7. és 8. évfolyamos tanulók válaszait. Ezeket az értékeket összevetettük a kutatócsoportunk által kidolgozott Hibafelismerési Modell (Error Recognition Model, ERM) hatékonyságvizsgálatának előtesztjén résztvevő 153 fő 9. osztályos középiskolai tanuló eredményeivel (Sebestyén et al. 2021). A mérések azt mutatják, hogy az évfolyamok önértékelése szorosan együtt mozog, nincs különbség a tanulók önértékelése között annak ellenére, hogy egy évnyi informatikatanulás van közöttük. A 7. évfolyamos tanulók 40,85%-a jelölte a legmagasabb, 5. jártassági szintet, míg az egy évfolyammal magasabb tanulók 37,80%-a választotta ezt a szintet. A 7. évfolyam magabiztossága egyértelműen utal a Dunning-Kruger effektus jelenlétére (Kruger & Dunning 1999). Az önértékelés és a tanulók gyakorlati tudása közötti eltérés fordítottan arányos (4. ábra). Az eltérés mutatja, hogy a hagyományos módszerekkel tanulók a témakört a szoftver felületén történő navigációval azonosítják.



4. ábra A mini kompetencia teszt szövegkezelés jártassági szintje és az ERM vizsgálat gyakorlati eredményei 0–5 Likert-skálán, a két vizsgálati eredmény összehasonlítása.

A fájlkezeléshez hasonlóan, a tanulók a mini-kompetencia tesztben jelölték, hogy az iskolában tanultak-e szövegkezelést. A tanulók 5,90%-a válaszolta, hogy nem tanulta iskolában a témakört. A mini kompetencia teszt egy másik kérdéscsoportja arra kereste a választ, hogy a tanulók milyen tevékenységeket folytatnak az informatika órákon. A tanulók 77,51%-a jelölte a szöveg gépelését, ezt követően 70,46% a játékot, szabadfoglalkozást, harmadikként 69,12% a táblázatok gépelését (Nagy et al. 2019). Elkésztető szembe-sülni azzal, hogy a legelterjedtebb tevékenységek közé a gépelés és a játék tartozik az informatikaórákon. A gépelés mint az informatikaóra célja, az egyik legfelelősebb és haszontalanabb tevékenység. Minden tanuló eltérő sebességgel gépel, ráadásul nem fejleszt egyetlen olyan képességet sem, ami a nevelési és oktatási célok között megjelenik. A játék és szabadfoglalkozás a második leginkább jellemző a tevékenységek között. Gyakorló informatikatanárként az egyik legnehezebb feladatomb volt, hogy a tanulókkal megérttessem, hogy az informatika nem egyenlő a játékkal, azonban ez még mindig sok helyen elterjedt. Ezzel összhangban van egy korábbi mérés eredménye (Sebestyén 2014), ahol egy kérdőíves felmérésben 41 informatikatanár vett részt. A tanárok 40%-a engedélyezte a játékot a tanórákon különböző mértékben (5–15 perc), emellett elégedetlenek voltak a kerettantervek (OFI 2012b) órakeretével és a tananyag teljesíthetőségével. Ezzel ellentétben, a másik szakjuk esetében – jellemzően matematika, fizika, biológia – mindössze a tanárok 16%-a engedte a játékot, azt is csak elenyésző mértékben, maximum 5 perc. Ezek az eredmények megmagyarázzák, hogy hogyan lehetséges az, hogy a tanulók nem tanultak szövegkezelést, valamint hogy miért ennyire alacsony a témában szerzett jártasságuk.

4.3 Hibafelismerési Modell

A kutatócsoportunk a szövegkezelés esetében is hátrahagyja a felületi megközelítést, ezzel szemben a lassú gondolkodást igénylő sémaépítést és a sémák aktivizálását támogató gyors gondolkodás megfelelő kombinációját részesíti előnyben (Kahneman 2011; Sweller et al. 2011). A szövegkezelés esetében a módosított Pólya-féle koncepcióalapú problémamegoldási stratégiát alkalmazzuk (Pólya 1954), ahol megkülönböztetjük a meglévő dokumentumok elemzését és az új problémák megoldását. A dokumentumok

elemzését a hibakereséssel, diszkusszióval kezdjük, majd ezt követi a hibajavítás és a helyes formázás. Ez a megoldási megközelítés a programozásoktatásban már elterjedt, kutatások egyértelműen bizonyítják a hatékonyságát és pozitív hatását (Gould 1975; Freiermuth et al. 2008; Bell & Newton 2013; Panko 2013; Jerinic 2014; Gander 2014).

A Hibafelismerési Modell (Error Recognition Model, ERM) módszer alkalmazása során három szakaszt különböztetünk meg, attól függően, hogy milyen mértékben vonjuk be a számítógépet. Az első szakasz az unplugged, amely során a tanulók számítógép nélkül dolgoznak. Minden tanuló kap egy nyomtatott dokumentumot, ahol piros tollal keresik a nyomtatásban azonosítható hibákat. A nyomtatott lapon jelölik a hibák helyét, valamint magyarázatot fűznek hozzá, hogy miért helytelen. A hibák hatékony felismerését elősegíti a hibák kategorizálása, amely a természetes nyelvű szövegekhez köthető szabályrendszeren alapul. Ez alapján nyomtatott dokumentumokban a következő hibakategóriák azonosíthatók: szintaktikai, szemantikai, tipográfia (Csernoch 2019, Sebestyén et al. 2021).

A második szakasz a semi-unplugged, ahol mind a nyomtatott, mind a digitális dokumentumra szükség van. A tanulók megnyitják a dokumentum digitális verzióját egy szövegszerkesztő program segítségével, és bekapcsolják a minden látszik gombot, ami megjeleníti a nyomtatásban nem látszódo karaktereket. A piros tollat, kékre cserélve tovább folytatják a hibák keresését, azonban most már olyan hibákat keresnek, amelyek csak a digitális dokumentumban azonosíthatók. A digitális dokumentumban ismerhetők fel a tördelési, a formázási és a stílus hibák (Csernoch 2019).

A harmadik, egyben utolsó szakasz a plugged-in, amikor a tanulók csak a számítógéppel dolgoznak és kijavítják a dokumentum hibáit, annak érdekében, hogy egy hibamentes és jól-szerkeszthető, jól formázható dokumentumot kapjanak végeredményül. A hibajavítást minden esetben követi egy negyedik lépés, amelyben, továbbra is a kutatásalapú megközelítést alkalmazva, megtörténik a dokumentum helyes formázása.

A módszer előnye, hogy a problémamegoldási folyamatot részekre bontja (unplugged, semi-unplugged és plugged-in), amelyek során csökkenthető a felismerhető hibák száma, és a megoldás során alkalmazott eszközök használata. Ennek következtében a módszer teret enged a tanulók figyelmének

irányítására, nem vonja el a számítógép mint eszköz a fókuszot (Kirschner 2002, Wolfram 2020). Ezentúl a hibák felismerése és magyarázata segíti a megértési folyamatot, és rögzíti a helyes megoldásokat, módszereket, továbbá támogatja a helyes terminológiahasználatot.

5. Sprego a táblázatkezelési ismeretek oktatásához

5.1 Széles körben elterjedt táblázatkezelői megközelítések

A táblázatkezelők programok a természetes nyelvű tartalom mellett, mesterséges nyelvet is használnak az adatfeldolgozás során. A mesterséges nyelv előnye, hogy a fordítóprogram képes ellenőrizni, figyelmeztetni a felhasználót, ha hiba szerepel a kódban. A futtatás során a fordítóprogram ellenőrzi a bevitt képletet, azonban ez csak a szintaktikai helyességet képes ellenőrizni, szemantikai elemzést nem végez. Általánosságban elmondható, hogy táblázatkezelői környezetben a felhasználókban nem alakult ki az igény a megoldások helyességének ellenőrzésére, ennek a következménye, hogy a táblázatok 86%-a tartalmaz valamilyen hibát (Panko 2008). Azokban az esetekben, amikor a táblázatkezelés gazdasági, pénzügyi területeken van jelen, egy helytelen hivatkozás, egy hibás képlet komoly anyagi veszteségeket okozhat (Eusprig 2020).

A széles körben elterjedt gyakorlatnak megfelelően, táblázatkezelés oktatása, a szövegkezeléshez hasonlóan felületi, a szoftverkörnyezetben történő navigáláson alapul, kezdve a program és a dokumentum ablak részeivel. A tankönyvi feladatok elsősorban 4–5 rekordból álló begépeltetett táblázatban történő feladatmegoldásból áll. A tanulók jogosan nem értik, hogy miért írjanak bármilyen számítási képletet, ha a megoldás szemmel kinézhető abból a 4–5 sorból. Tovább nehezíti a tanulási-tanítási folyamatot, hogy a kerettantervek és a hozzájuk igazodó tankönyvek közel 100 táblázatkezelői függvény ismeretét várják el (Csernoch et al. 2014). A rendelkezésre álló órakeret, a függvények mennyisége, a hozzájuk tartozó argumentumok sorrendje, ezek szintaktikája, szemantikája, értelmezési tartománya és értékkeszlete olyan adatmennyiség, amely megtanulhatatlan. A tankönyvben javasolt függvények száma azért közelíti a 100-at, mert egy-egy probléma megoldására alkalmasak csak, így ahány probléma, annyi függvény ismeretere van szükség.

5.2 Sprego programozás és adatkezelés

A Sprego (Spreadsheet Lego) egy koncepcióalapú programozási megközelítés, amely a táblázatkezelői környezetek funkcionális nyelvét használja kódolási környezetként. Korábbi mérések igazolják, hogy a Sprego megközelítés alkalmas táblázatkezelői, adatfeldolgozási, valamint algoritmizálási és programozási ismeretek tanítására is. Az eddig elvégzett vizsgálatok egyértelműen bizonyítják, hogy a Sprego programozási megközelítéssel lényegesen hatékonyabban tanítható a táblázatkezelés, mint a széles körben elterjedt felületi megközelítési módszerekkel (Csapó et al 2020, Csernoch et al. 2021, Sebestyén et al. 2019).

A Sprego módszer egyik nagy előnye, hogy 12 általános célú függvényt használ, amelyek problémától függően összeépíthetőek (Csernoch 2014). A Sprego által használt 12 függvény lefedi a középiskolai és az érettségi követelményeket, így beilleszthető a tantervekbe (Csernoch & Biró 2015b). Bizonyításra került továbbá, hogy Sprego az emelt szintű informatika érettségi programozási feladatainak megoldásához legalább olyan hatékonyan alkalmazható, mint a hagyományos imperatív és objektumorientált programozási nyelvek (Glevitzky & Csernoch 2021).

A problémamegoldási és algoritmizálási folyamatok megértését a kutatócsoportunk által kidolgozott sok-sok unplugged (3D nyomtatott egymásba ágyazható babák, origami hajók, kereskedelmi forgalomban kapható hordókészlet, számtalan egyéb kiegészítő) és semi-unplugged (2D és 3D oktatószoftver) eszköz segíti (Csapó & Sebestyén 2015, 2017, Sebestyén et al. 2018, 2019).

6. Hipotézisek

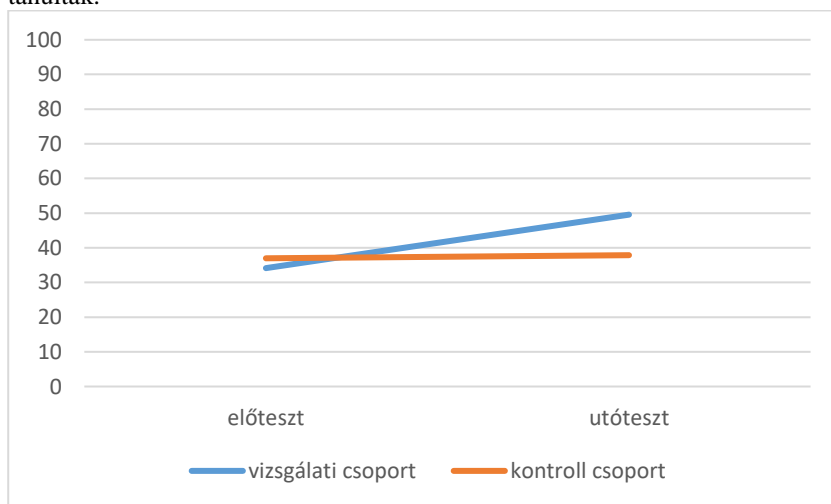
[T1.] A séma és algoritmusalapú módszertanok (WDC, ERM, Sprego) szignifikánsan hatékonyabbak, mint a hagyományos, felületalapú megközelítések.

A kutatócsoportunk által kidolgozott módszerekhez hatékonyságvizsgálatokat tervezetünk és végeztünk, amelyek menete minden esetben megegyezett. A vizsgálatban résztvevő tanulók a tanulási folyamat előtt és után tesztet

írtak, melynek eredményeit összehasonlítottuk. A sémaalapú módszer hatékonyságának bizonyításához a saját csoportjainkon túl, a közoktatásban elterjedt, hagyományos felületi megközelítést alkalmazó módszerekkel tanuló kontrolcsoportokat használtunk. A kontroll csoportok bevonásához minden esetben szükség volt kutatócsoporton kívüli pedagógusokra. Néhány esetben egy-egy vizsgálat vagy csoport eredménye értékelhetlenné vált a kollégák visszalépése miatt. Emiatt és az informatika órákon alkalmazott csoportbontás miatt (fél osztály) alacsony a mintában szereplő tanulók száma. Az elő- és az utótesztek összehasonlítása során azon tanulók eredményeit vettük alapul, akik mindkét felmérésben részt vettek, tehát nyomon követhető az önmagukhoz történő fejlődés.

Fájlkonverzió – WDC

A WDC módszer hatékonyságvizsgálatát a 2018/2019-es tanévben valósítottuk meg. A vizsgálatban a középiskolát megkezdő 9. évfolyamos tanulók vettek részt. A vizsgálati csoportot 30 fő alkotta, akik a kutatócsoport által fejlesztett WDC módszerrel sajátították el a fájlkezelés témakört, míg a kontroll csoport 79 főből állt, akik hagyományos kerettantervre épülő módszerrel tanultak.



5. ábra A WDC módszer hatékonyságvizsgálatának eredménye.

Az előteszt során a csoportok hozott tudása között nem volt szignifikáns eltérés ($p=0,0607$) (5. ábra). Az oktatási folyamat végén a vizsgálati csoport szignifikánsan magasabb eredményt ért el a kontroll csoporthoz képest ($p=0,0000$). Az eredmény igazolta a WDC módszer hatékonyságát, azonban meglepetést okozott a kontroll csoport eredménye. A csoport az utóteszten (37,88%) egyáltalán nem mutatott fejlődést az előteszthez (38,02%) képest (5. ábra). Ez nem csak a hagyományos módszerek hatékonyságát kérdőjelezi meg, hanem a tényleges tanórai tevékenységet helyettesítő gépelés, játék, szabadfoglalkozások jelenlétét is.

Hibafelismerés és keresés – ERM

Az ERM módszer hatékonyságvizsgálatát a 2017/2018-as tanévben való-sítottuk meg. A vizsgálatban 9. évfolyamos tanulók mellett 7. évfolyamos tanulók vettek részt, akik szintén első évüket kezdték a középiskolában, 6 osztályos gimnáziumi képzésben. A kerettantervek szerint a 9. évfolyamos tanulók már elsajátították a szövegszerkesztés „alapjait” a korábbi évek során. A felmérésben összesen 153 tanuló vett részt (7. évfolyamos vizsgálati (7V) 34 fő, 9. évfolyamos vizsgálati (9V) 69 fő, 9. évfolyamos kontroll (9K) 50 fő) (3. táblázat). A tesztelési folyamat megegyezett a módszer ismertetésekor bemutatott három szakasszal, unplugged, semi-unplugged és plugged-in. Az előteszten az azonos évfolyamos tanulók tudása között nem volt szignifikáns eltérés. A párosított tesztek eredményei alapján minden csoport szignifikáns mértékben fejlődött az előteszthez képest (3. táblázat). Az azonos évfolyamos tanulók fejlődési üteme azonos volt. A legnagyobb mértékben a 7V csoport fejlődött, akik az ERM módszerrel tanultak, annak ellenére is, hogy az utóteszten a szintaktikai hibákat elenyésző mértékben ismerték fel. A nyomtatásban észrevehető hibák tipográfiai, tördelési és szintaktikai kategóriába sorolhatók, ez utóbbi elsősorban nyelvhelyességi hibákat jelent, amiknek felismerése anyanyelvi ismereteket igényel. Az alacsonyabb évfolyamos tanulók kiugró fejlődése arra utal, hogy az ERM módszer már a szövegkezelés első oktatásakor hatékony(abb)an alkalmazható. Az ERM módszer problémamegoldáson és sémaépítésen alapuló szerkezete biztosítja a hosszútávú tudás kialakítását, valamint hatékonyabb eredményeket produkál,

mint a hagyományos megközelítések – amivel a 9. évfolyamos tanulók a felmérést megelőző években már több éven keresztül foglalkoztak.

	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%

3. táblázat Az ERM hatékonyságvizsgálatának eredménye.

Sprego

A Sprego módszer hatékonyságát már korábbi mérések során sikerült bizonyítanunk (Csapó et al. 2019). További közös munkánk eredményeként megvizsgáltuk a módszer hosszútávú hatását, amelyhez egy késletetett utótesztet terveztünk. A beavatkozást követően kb. egy évvel később ismét teszteltük a tanulók tudását (Csapó et al 2021).

	vizsgálati csoportok				kontroll csoportok		
	18 fő	18 fő	17 fő	15 fő	7 fő	13 fő	7 fő
	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

4. táblázat Sprego késletetett utóteszten résztvevő csoportok létszáma és táblázatkezeléssel töltött óraszám az előző évből.

A minta kiválasztásakor olyan csoportokat válogattunk be a vizsgálatba, akik egy évvel ezelőtt tanultak táblázatkezelést. A mintát 95 tanuló alkotta.

A vizsgálati csoport 8. és 10. évfolyamos tanulókból állt, míg a kontroll csoport 10. évfolyamos tanulókból (4. táblázat). Az egyes csoportok a vizsgálatot megelőző tanévben eltérő óraszámú tanultak táblázatkezelést a különböző képzési típusok miatt (4. táblázat).

A késleltetett utóteszt során a tanulók adatelemzési, algoritmus- és sémaalkotási, valamint függvényalkotási és értelmezési képességét vizsgáltuk hat feladat segítségével.

A vizsgálati csoport egy feladat kivételével minden esetben szignifikánsan magasabb eredményt ért el, mint a kontroll csoport (5. táblázat). A legmagasabb eredményt az e10_1 vizsgálati csoport érte el 55,08%-kal, a leggyengébben a kontroll csoport c10_3 csoportja teljesített. Érdemes figyelmet fordítani az alacsonyabb évfolyamos tanulók eredményeire is, ahol az e8_2 csoport hasonlóan magas eredményeket ért el, mint a vizsgálati csoport 10. évfolyamos tanulóit. A vizsgálati csoport „leggyengébben” teljesítő e8_1 csoportja is szignifikánsan jobban teljesített, mint a tőle két évvel idősebb, hagyományos módszerekkel tanult kontroll csoportok (c10_1, c10_2, c10_3). A kontroll csoport c10_1 és c10_2 osztálya a magas óraszám ellenére sem képes megközelíteni a Sprego módszerrel tanított vizsgálati csoportok eredményeit.

	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

5. táblázat A Sprego hosszútávú tudás mérésének eredménye csoportonként és feladatonként részletezve (Csapó et al. 2021)

A WDC és az ERM módszer hatékonyságvizsgálata és a Sprego módszer késleltetett utótesztjének eredménye igazolja a hipotézist. Mindhárom módszer esetében szignifikáns mértékben fejlődtek a vizsgálati csoportok tagjai. A fiatalabb évfolyamok tanulói esetében a hatékonyság fokozottabb, ezt az ERM 7V csoportjának kimagasló fejlődése és a Sprego késleltetett utóteszt 8. évfolyamos csoportjainak eredményei igazolják, akik az első alkalom után magasabb eredményt értek el, mint a hagyományos módszerrel több éven át tanuló kontroll csoport tagjai.

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

A kutatócsoport által alkalmazott problémamegoldáson alapuló módszerek elsajátítása során a tanulók számítógépes gondolkodása és algoritmizálási képessége fejlődik, melyet iskolán kívüli problémák esetében is hatékonyan tudnak alkalmazni (Merriënboer & Sweller 2005; Skemp 1971). A módszerek hatékonysága a problémamegoldáson és az autentikus tartalmakon túl, abban áll, hogy alapoz a tantárgyon belüli és tantárgyközi kapcsolatokra és a tudástranszferelemekre. A fájlkezelés oktatása során alkalmazott fájlkonverziós módszer elsajátítása során a tanulók megismerkednek különböző adatmanipulációs műveletekkel, amelyeket jellemzően szöveg- és táblázatkezelői környezetben hajtanak végre. Az adatmanipulációs tevékenységek megalapozzák és elősegítik ezeknek a témaköröknek a későbbi megtanulását. Egy újabb témakör bevezetése nem egy új fejezetet jelent, ahol el lehet felejtetni minden korábbi ismeretet, mert az nem szükséges az új ismeret elsajátításához. Ezzel szemben, az általunk kidolgozott módszerek kiemelten alapoznak a tudástranszferelemek építésére és aktiválására. A szöveg- és táblázatkezelés oktatása során nélkülözhetetlen a fájlkezelés oktatásánál megszerzett ismeret a hatékony problémamegoldáshoz. Ugyan különböző módszerek tartoznak az egyes témakörökhöz, azonban mindegyik a tanulók sémákban való gondolkodását segíti, és lehetővé teszi problémamegoldási stratégiák elsajátítását, melyek lépéseire a tanulók már a fájlkezelés tanítása során szert tesznek. Idővel változik a szoftverkörnyezet, bővül a felhasználható eszközök köre, valamint a felhasználók tudása, amit kreatív módon tudnak beépíteni és felhasználni a problémamegoldás során, iskolai környezetben és azon túl is.

A kerettantervekre épülő hagyományos módszerek az informatikai témaköröket egymástól elszeparáltan tanítják. A tanulók a témakör befejezése után nem használják a megszerzett ismereteket mindaddig, amíg a spirális oktatási rendszer miatt ismételen elő nem kerül a téma a következő tanévekben, ha előkerül egyáltalán. Ennek a felépítésnek az egyik legnagyobb hátránya, hogy az egy-egy témakörre szánt 3–10 órakeret nem elegendő a kerettantervekben megfogalmazott ismeretek átadására, gyakorlására, rendszerezésére és rögzítésére. Ezzel szemben a sémaépítésen és problémamegoldáson alapuló módszereink a tanulók gondolkodásmódját fejlesztik, amelynek segítségével a tanulók hatékonyan el tudják dönteni, hogy a megoldáshoz milyen eszközt érdemes alkalmazni.

A kutatócsoport által alkalmazott módszerek hatékonyságvizsgálatát külön-külön végeztük el, így jelenleg nincs statisztikai bizonyítékunk a módszerek egymást felerősítő hatásáról. Ezek a vizsgálatok további, több éven átívelő kutatásokat igényelnek, amelyek messze túlmutatnak a PhD keretein. Jelenleg saját tapasztalataink és a kutatásban résztvevő kollégák és tanárszakos hallgatók véleménye alapján a megfogalmazott tézis elfogadható. A rendelkezésre álló adatok alkalmasak arra, hogy előkészítsünk egy több éven át tartó kutatást a tézis bizonyítására.

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

A mini kompetencia országos felmérésben 8880 fő 7–10. évfolyamos tanuló vett részt. A felmérés a 2017/2018-as tanév végén került lebonyolításra, így az általános iskolából hozott ismeretek vizsgálatához a 7. és 8. évfolyamos tanulók válaszait vettük alapul (Nagy 2018). A tanulók önértékelés alapján jelölhették meg a megadott informatikai témakörök esetében, hogy milyen magas jártassági szinttel rendelkeznek. A fájlkezelés és a szövegkezelés témakörökre érkezett válaszok kerültek kiértékelésre, melynek eredményét összehasonlítottuk a WDC és az ERM módszer hatékonyságvizsgálatának előteszt eredményeivel. Ezeket a tesztek 9. évfolyamos középiskolai tanulók töltötték ki a témakör tanulása előtt, tehát tudásuk annyi, amennyit az általános iskolában elsajátítottak. Az előtesztek mindkét informatikai tudást

mérő (WDC és ERM) vizsgálat során azt bizonyították, hogy a tanulók tudása hiányos, több esetben tévhitekkel terhelt, a feladatokon alacsony eredményeket értek el – WDC 9. évfolyam 32,34%, ERM 7. évfolyam: 20,29%; 9. évfolyam: 28,62%). Az azonos ismeretet igénylő feladatok esetében a tanulók válaszaik véletlenszerűek voltak, ami az ismeretek rendszerezettségének hiányára utal. Ezek a félreértések és eredmények nem igazolják a tanulók által jelölt jártassági szintet, azonban igazolják a tézist, mely szerint a tanulók nem, vagy alig tudnak támaszkodni az általános iskolából hozott ismeretekre.

[T4.] A tanulók az informatikai ismereteiket az önértékelés során túlértékelik

A mini kompetencia teszt eredményeit összehasonlítottuk a WDC és az ERM módszer hatékonyságvizsgálatának előtesztjeivel. Az összehasonlítások mindkét témakör esetében, fájlkezelés és szövegkezelés, egyértelműen bizonyítják, hogy a tanulók gyakorlati tudása jóval alatta maradt az önértékelésen adott értékeknek (3. ábra, 4. ábra). Minden esetben egyértelműen nyomon követhető a Dunning-Kruger (Kruger & Dunning 1999) effektus, mivel az alacsonyabb évfolyamos tanulók magasabb jártassági szintet jelölnek meg. Ezek az eredmények mutatják, hogy az informatikai kompetenciák mérésére a tanulók alacsony tudásszintje miatt az önbevalláson alapú értékek nem adnak valós képet. A tanulók körében végzett vizsgálatok alapján elgondolkodtató, és további vizsgálatok szükségesek annak megválaszolására, hogy más életkorban hogyan alakul az önértékelés és a valódi tudás egymáshoz való viszonya.

7. 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 témakör 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 módszerek hatékonyságvizsgálata során bevont tanulói csoportok létszáma alacsony, így a mérések szélesebb körben való elvégzését kívánom megvalósítani, amit elsősorban több intézmény, pedagógus és tanuló bevonásával tartok megvalósíthatónak. Mindkét mérést jelenléti oktatás formájában szeretném megvalósítani, ugyanis egyelőre az online oktatás és mérésértékelés eredményeinek megbízhatósága megkérdőjelezhető.

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 szemléletváltásában.

8. Irodalomjegyzék

- Bánné Mászáros, A., Csintalan, T. & Lakosné Makár, E. (2008). Informatika 5. Nemzeti Tankönyvkiadó, Budapest.
- Balázsi, I. & Ostorics, L. (2011). PISA 2009: Digitális szövegértés. Olvasás a világhálón. Oktatási Hivatal. Budapest
- Bártfai, B. (2011). Számítógéphasználat mindenkinek. BBS-INFO Könyvkiadó és Informatikai Kft., Budapest.
- Bell, T. & Newton, H. (2013). Unplugging Computer Science. In Improving Computer Science Education. (Eds.) Djordje M. Kadijevich, Charoula Angeli, and Carsten Schulte. Routledge. pp. 66-81.
- Ben-Ari, M. (1999). Bricolage Forever!, PPIG 1999. 11th Annual Workshop. 5–7 January 1999, Computer-Based Learning Unit, University of Leeds, UK, 1999, pp. 1-5. Letöltve 2020.10.10. <https://ppig.org/files/1999-PPIG-11th-benari.pdf>
- Chmielewska, K. & Gilányi, A. (2019). Computer Assisted Activating Methods in Education. 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Naples, Italy, 2019, pp. 241-246.
- 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. Letöltve 2021.05.10. <https://link.springer.com/content/pdf/10.1007/s10639-019-10024-2.pdf>
- Csapó, G. & Sebestyén, K. (2015). Oktatóprogram a Sprego táblázatkezelő módszerhez. INFODIDACT 2015: Informatika Szakmódszertani Konferencia. 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. http://www.tojet.net/special/2017_10_1.pdf
- 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. (2014). Programozás táblázatkezelői függvényekkel. Műszaki Tankönyvkiadó, Budapest.
- 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, 2019, pp.147-152.
- Csernoch, M. & Biró, P. (2015a) Wasting Human and Computer Resources. International Journal of Social Education Economics and Management Engineering, 9(2), pp. 555-563.
- Csernoch, M. & Biró, P. (2015b). Sprego helye az informatika tantervekben. Infodidact 2015. Letöltve 2021.07.10. <https://people.inf.elte.hu/szlavi/InfoDidact15/Manuscripts/CsMBP.pdf>
- 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. Abari, K. & Máth, J. (2014). Programozásorientált táblázatkezelői függvények. XIV. Országos Neveléstudományi Konferencia: Oktatás és nevelés – gyakorlat és tudomány : Tartalmi összefoglalók. Debreceni Egyetem Neveléstudományok Intézete (2014), pp. 463-463.
- Csernoch, M.; Biró, P. & Máth, J. (2021). Developing computational thinking skills with algorithm-driven spreadsheeting. IEEE Access. 9. pp. 153943-153959.
- 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.
- Dancsó, T. & Korom, P. (2013). Informatika 9-10. – A gimnáziumok számára. Nemzedékek Tudása Tankönyvkiadó. Budapest.
- ECDL Foundation (2019). ECDL vizsgapéldatár. Számítógépes-alapismeretek. Syllabus 1.0. Neumann János Számítógép-tudományi Társaság.
- Eusprig (2020). Horror Stories. Letöltve: 2020.10.10. <http://eusprig.org/horror-stories.htm>

- Freiermuth, K., Hromkovic, J. & Steffen, B. (2008). Creating and Testing Textbooks for Secondary Schools. In: Mittermeir R.T., Syslo M.M. (eds) Informatics Education - Supporting Computational Thinking. ISSEP 2008. Lecture Notes in Computer Science, vol 5090. Springer, Berlin, Heidelberg. pp. 1-13.
- Gander, W. (2014). Informatics and General Education. In: Gülbahar Y., Karataş E. (eds) Informatics in Schools. Teaching and Learning Perspectives. ISSEP 2014. Lecture Notes in Computer Science, vol 8730. Springer, Cham. pp. 1-49.
- Glevitzky, K. & Csernoch, M. (2021). Az emelt szintű informatika érettségi programozás feladatainak megoldása Spregoval. Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2021.
- Gould, J. (1975). Some psychological evidence on how people debug computer programs. *International Journal of Man-Machine Studies*, (7) 1, pp. 151-182.
- Hom, S. N. H., Ibrahim H. H., Ibrahim, A., Mokhsin, M. & Talib, C. A. (2020). Assessment of using EZ-prog: an easy color schematic model for programming problem solving. *Asean Journal of Teaching & Learning in Higher Education* Vol 12., No 1. pp. 31-41.
- 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*, (5) 6. pp. 53-59.
- 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 - Teaching and Learning in the New Digital Landscape*. SAGE Publications. Thousand Oaks, United States.
- Kahneman, D. (2011). *Thinking, Fast and Slow*. New York: Farrar, Straus and Giroux
- 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), 75-6.

Kirschner, P. & De Bruyckere, P. (2017). The myths of the digital native and the multitasker, *Teaching and Teacher Education*. 67 (2017), pp. 135–142.

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 Soci-al Psychology*, 1999, 77 (6), pp. 1121-1134.

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

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

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

Merriënboer, J. J. G. van & Sweller, J. (2005). Cognitive Load Theory and Complex Learning: Recent Developments and Future Directions. *Educational Psychology Review*, 2005, 17(2), pp. 147-177.

Nagy, T. K. (2018). 7–10. évfolyamos tanulók tudástranszfer alapú számítógépes gondolkodásának elemzése reprezentatív felmérés alapján. Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2018. Letöltve 2020.05.10. <http://hdl.handle.net/2437/277191>

Nagy, T. K. & Csernoch, M. (2018). The paradox of the hungarian frame curricula in informatics. *The Turkish Online Journal of Educational Technology*, INTE 2018. pp. 783-795. Letöltve 2020.09.12. http://www.tojet.net/special/2018_12_3.pdf

Nagy, T. K.; Csernoch, M. & Máth, J. (2019) 7–10. évfolyamos tanulók tudástranszfer alapú számítógépes gondolkodásának elemzése. ADA 2019 Konferencia, Debrecen. közlésre elfogadva

OECD (2011). *PISA 2009 Results: Student On Line: digital technologies and performance (Volume VI)*. Letöltve 2020.02.10. <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: 2020.07.10. <http://www.nefmi.gov.hu/kozoktatas/tantervek/oktatasi-kulturalis>
- OFI (2012a). NAT 2012. 110/2012. (VI. 4.) Korm. rendelete a Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról. Letöltve 2020.12.10. 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: 2020.11.19. https://www.oktatas.hu/koznevelas/kerettantervek/2012_nat
- 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: 2020.11.15. <https://magyarkozlony.hu/dokumentumok/3288b6548a740b9c8daf918a399a0bed1985db0f/megtekintes>
- 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. (2013). The Cognitive Science of Spreadsheet Errors: Why Thinking is Bad. Proceedings of the 46th Hawaii International Conference on System Sciences, January 7-10, 2013, Maui, Hawaii.
- Panko, R. R. & Aurigemma, S. (2010). Revising the Panko-Halverson taxonomy of spreadsheet errors. Decis. Support Syst. 49, 2., pp. 235-244.
- Papp, P. & Csernoch, M. (2018). A táblázatkezelés is problémamegoldás? Infodidact 2018. Letöltve 2020.11.10. <https://people.inf.elte.hu/szlavi/InfoDidact18/Manuscripts/PPCsM.pdf>
- Pólya, G. (1954). How To Solve It. A New Aspect of Mathematical Method. Princeton University Press, Princeton, New Jersey.
- Prensky, M. (2001). Digital Natives, Digital Immigrants, From On the Horizon (MCB University Press, Vol. 9 No. 5, October 2001)

- Sebestyén, K. (2014). Te is lehetsz informatikatanár (?). INFODIDACT 2014. 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 Rep-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
- 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
- Sebestyén, K., Csapó, G., Csernoch, M. & Aradi, B. (2021) Error Recognition Model: End-user Text Management. Acta Polytechnica Hungarica. Accepted.
- Skemp, R. R. (1971). The Psychology of Learning Mathematics. New Jersey: Lawrence Erlbaum Associates.
- Szabados, S. (2009). Digitális bennszülöttek. Oktatás-informatika I. évfolyam, 2009. 1. szám, pp. 19-23.
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning, Cognitive Science, 12, pp. 257-285.
- Sweller, J., Ayres, P. & Kalyuga, S. (2011). Cognitive Load Theory. Springer.
- U. S. Department of Education (2018). The Condition of Education 2018. NCES 2018-144. IES. Letöltve 2020.10.12. <https://nces.ed.gov/surveys/icils/icils2018/pdf/2019164.pdf>
- Varga, P., Jeneiné Horváth, K., Reményi, Z., Farkas, Cs., Takács, I., Siegler, G. & Abonyi-Tóth, A. (2020). Digitális kultúra 9. Oktatási Hivatal, Budapest. Letöltve 2021.02.18. https://www.tankonyvkatalogus.hu/pdf/OH-DIG09TA__teljes.pdf
- Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), pp 33-35.
- Wolfram, C. (2020). The Math(s) Fix: An Education Blueprint for the AI Age. Wolfram Media, Inc.



Nyilvántartási szám: DEENK/484/2021.PL
Tárgy: PhD Publikációs Lista

Jelölt: Sebestyén Katalin
Doktori Iskola: Informatikai Tudományok Doktori Iskola
MTMT azonosító: 10054523

A PhD értekezés alapjául szolgáló közlemények

Idegen nyelvű tudományos közlemények hazai folyóiratban (2)

1. **Sebestyén, K.**, Csapó, G., Csernoch, M., Aradi, B.: Error Recognition Model: End-user Text Management.
Acta Polytech. Hung. "Accepted by Publisher" (-), 1-20, 2021. ISSN: 1785-8860.
IF: 1.806 (2020)
2. **Sebestyén, K.**, Csapó, G., Csernoch, M., Abari, K.: Measuring the algorithmic skills of students working with low- and high-mathability programming approaches.
Acta Polytech. Hung. "Accepted by Publisher" (-), 1-20, 2021. ISSN: 1785-8860.
IF: 1.806 (2020)

Idegen nyelvű tudományos közlemények külföldi folyóiratban (1)

3. Csapó, G., **Sebestyén, K.**, Csernoch, M., Abari, K.: Case study: developing long-term knowledge with Sprego.
Educ. Inf. Technol. 26 (1), 965-982, 2021. ISSN: 1360-2357.
DOI: <http://dx.doi.org/10.1007/s10639-020-10295-0>
IF: 2.917 (2020)

Magyar nyelvű konferencia közlemények (1)

4. **Sebestyén, K.**: Te is lehetsz informatikatanár (?).
In: INFODIDACT 2014. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika Alapítvány, Zamárdi, 1-7, 2014. ISBN: 9789631206272

Idegen nyelvű konferencia közlemények (4)

5. **Sebestyén, K.**: Students' Knowledge in File Management.
In: Proceedings of the 11th International Conference on Applied Informatics (ICAI 2020) Ed.: by Gergely Kovásznai, István Fazekas, Tibor Tómacs, CEUR, Eger, 296-308, 2020, (CEUR Workshop Proceedings, ISSN 1613-0073 ; 2650.)
6. **Sebestyén, K.**, Csapó, G., Csernoch, M.: Introduction to Algorithmic Based Data Management in Spreadsheet Environment.
Turk. Online J. Educ. Technol. 2019 (Spec.), 178-185, 2019. EISSN: 1303-6521.





7. Biró, P., Csernoch, M., **Sebestyén, K.**, Gombos, E.: Algorithmic Skills Vs. Time Spent On Computers And Mobile Devices.
Turk. Online J. Educ. Technol. 1 (Spec.), 133-141, 2018. ISSN: 2146-7242.
8. **Sebestyén, K.**, Csapó, G., Csernoch, M.: Visualising Sprego Inequality Problems With 2D Representations.
Turk. Online J. Educ. Technol. Special Issue for INTE-ITICAM-IDEK 2018, 888-898, 2018.
EISSN: 1303-6521.

További közlemények

Magyar nyelvű konferencia közlemények (1)

9. Csapó, G., **Sebestyén, K.**: Oktatóprogram a Sprego táblázatkezelő módszerhez.
In: INFODIDACT 2015. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika Alapítvány, Zamárdi, 1-18, 2015. ISBN: 9789631238921

Idegen nyelvű konferencia közlemények (1)

10. Csapó, G., **Sebestyén, K.**: Educational software for the Sprego method.
Turk. Online J. Educ. Technol. Special Issue for INTE 2017, October, 986-999, 2017. EISSN: 2146-7242.

Magyar nyelvű absztrakt kiadványok (1)

11. Csapó, G., **Sebestyén, K.**: Bevezetés az algoritmus alapú adatkezelésbe táblázatkezelői környezetben: Digitális Témahét tapasztalatai.
In: ADA 2019 : Rövid absztraktok, Debrecen Egyetemi Kiadó, Debrecen, 8, 2019.

A közlő folyóiratok összesített impakt faktora: 6,529

A közlő folyóiratok összesített impakt faktora (az értekezés alapjául szolgáló közleményekre) 6,529

A DEENK a Jelölt által az IDEa Tudóstérbe feltöltött adatok bibliográfiai és tudománytermetriai ellenőrzését a tudományos adatbázisok és a Journal Citation Reports Impact Factor lista alapján elvégezte.



Debrecen, 2021.11.04.

Short thesis for the degree of doctor of philosophy (PhD)

Developing students' schema- and knowledge-transfer-based computational thinking skills through teaching informatics applications

by Katalin Sebestyén
Supervisor: Mária Csernoch



UNIVERSITY OF DEBRECEN
Doctoral School of Informatics

Debrecen, 2021

1. Introduction

It is not only students and office workers who must have IT application skills, since the digital world is part of our everyday life. We are surrounded by biotical applications and problems associated with taking part in digital education and advanced studies, keeping track of students' achievements in the digital diary, keeping contact with others, managing official documents, etc. The first place where we encounter the guided and mastered use of the acquisition of ICT applications and the development of fundamental computational thinking skills and abilities is public education, where effective problem-solving methods and strategies can be developed at a young age. Those students who are born into and grow up in in the digital era are called digital natives (Prensky 2001; Jukes & Dosaj 2006, Szabados 2009; Jukes et al., 2010). It is assumed that they speak and understand the language of digital technologies like they do their native language, and use such tools as smart devices, computers, game consoles, the Internet etc. It is also stated that the members of the digital generations are able to master the use of these devices using self-taught or trial-and-error approaches. Most them are also convinced that they do not need an ICT education, because they know everything. However, it has been proved that this is not so (Kirschner et al. 2006). Children learn how to talk from their parents, but no one questions the need to study language and literature in school, and the same is true for mathematics. The question arises: is there any difference between the need to study computational thinking compared to reading, writing, and arithmetic? The answer is that there is no difference. Students need ICT education, which is essential for developing fundamental skills in digital sciences and technology. The digital skill is/will be the fourth basic skill of the digital era, alongside Reading, wRiting, and aRithmetic (the 'three Rs') (Wing 2006). Consequently, students must learn the terminology and the procedures of ICT in order to be able to communicate, to solve problems effectively, and to see and understand the background to the subject.

In recent years several studies have proved that being a member of the digital generation does not automatically bring with it the mastering of high level digital literacy or computer thinking skills and abilities (Prensky 2001;

Kirschner & Bruyckere, 2017; U. S. Department of Education 2018). It is also proven that the everyday use of smart devices does not help the development of computational and algorithmic thinking abilities, especially not of computer problem-solving abilities (Csernoch & Biró 2019).

The importance of ICT education is also supported by the fact that 80–90% of text and spreadsheet documents are incorrect (Ben-Ari 1999; Panko 2008; Panko & Aurigemma 2010, Csernoch & Biró 2015). These errors cause serious financial losses, both to the authors and the business partners and companies involved (Eusprig 2020).

In 2009, students from 19 countries, including Hungary, took part in the PISA Student Online test (OECD 2011), which focused on the digital competence of the participants.

Level	Characteristics of tasks
5 or above	Tasks at this level typically require the reader to locate, analyse and critically evaluate information, related to an unfamiliar context, in the presence of ambiguity. They require generating criteria to evaluate the text. Tasks may require navigation across multiple sites without explicit direction, and detailed interrogation of texts in a variety of formats
4	Tasks at this level may require the reader to evaluate information from several sources, navigating across several sites comprising texts in a variety of formats, and generating criteria for evaluation in relation to a familiar, personal or practical context. Other tasks at this level demand that the reader interpret complex information according to well-defined criteria in a scientific or technical context.
3	Tasks at this level require that the reader integrate information, either by navigating across several sites to find well-defined target information, or by generating simple categories when the task is not explicitly stated. Where evaluation is called for, only the information that is most directly accessible or only part of the available information is required.

2	<p>Tasks at this level typically require the reader to locate and interpret information that is well-defined, usually relating to familiar contexts. They may require navigation across a limited number of sites and the application of web-based navigation tools such as drop-down menus, where explicit directions are provided or only low-level inference is called for. Tasks may require integrating information presented in different formats, recognizing examples that fit clearly defined categories</p>
below 2	

Table 1: A summary of the levels of proficiency in digital skills of the Pisa Student On Line 2009 (OECD 2011).

The Hungarian students' performance in digital reading, navigation and computer use did not reach the OECD average, and they finished third from bottom in the overall measurement (Figure 1). The results were evaluated based on the proficiency levels of the digital skills presented in Table 1.

Level 2 is considered the basic level. Below level 2 there is a place-holder reserved for cases in which knowledge items are not recognizable. Around half of Hungarian students were at level 2 (24.97%) or below 2 (26.82%); in other words they are digitally illiterate. In the next measurement, in 2012, the number of the students with lower digital competence increased (OECD 2015). In addition to the low level of the digital competence of Hungarian students, these measurements pointed out the large standard deviation of the results of Hungarian students when considering the best (5%) and the worst (5%) students (OECD 2011). The analysis also compared the scores between those students who use, and those who do not use, a computer at school. In this comparison, considering the whole sample, the largest negative correlation was found in the case of Hungarian students (Figure 1).

The students' low scores in the test and the relatively high number of ICT lessons caused this negative correlation, which draws attention to the inefficiency of Hungarian ICT education. These results were published in the OECD report in 2011 (OECD 2011).

■ Table VI.A ■
AN OVERVIEW OF PERFORMANCE IN DIGITAL READING, NAVIGATION AND COMPUTER USE

		Higher quality or equity than OECD average			At OECD average (no statistically significant difference)			Lower quality or equity than OECD average		
OECD		Digital reading performance	Gender difference in digital reading scores between boys and girls	Index of number of relevant pages visited (navigation skills)	Computer use at home			Computer use at school		
					Percentage of students who use a computer at home	Percentage difference between top and bottom quarters of the PISA index of economic, social and cultural status	Difference in digital reading scores between those students who use and those who do not use a computer at home	Percentage of students who use a computer at school	Percentage difference between top and bottom quarters of the PISA index of economic, social and cultural status	Difference in digital reading scores between those students who use and those who do not use a computer at school
	OECD average	499	-24	46.3	92.3	16.0	80	74.2	0.3	9
	Korea	568	-18	52.8	87.5	19.5	49	62.7	3.5	2.1
	New Zealand	537	-40	49.7	92.5	20.2	90	83.4	6.4	20
	Australia	537	-28	49.6	96.7	7.8	84	91.6	5.6	42
	Japan	519	-23	50.1	75.9	38.6	48	59.3	2.6	14
	Iceland	512	-30	47.5	99.1	1.2	74	79.5	5.1	22
	Sweden	510	-26	47.8	97.7	4.7	105	89.1	4.7	28
	Ireland	509	-31	47.4	93.2	10.9	60	62.9	0.4	-3
	Belgium	507	-24	47.7	96.9	9	102	62.8	-1.1	9
	Norway	500	-35	46.9	98.7	2.7	77	93.0	2.5	25
	France	494	-20	46.1	m	m	m	m	m	m
	Denmark	489	-6	47.2	98.8	2.8	79	93.0	1.8	6
	Spain	475	-19	44.2	93.6	14.4	78	65.5	-4.0	11
	Hungary	468	-21	41.6	91.8	23.6	102	69.3	-8.9	-27
	Poland	464	-29	42.0	92.1	22.9	84	60.6	-9.1	-8
	Austria	459	-22	43.3	98.2	3.7	94	84.1	-3.2	-6
	Chile	435	-19	37.7	73.2	60.3	69	56.8	-2.0	2

Figure 1. Results of the PISA Student On Line, highlighting the negative correlation between digital reading skills and school computer use of Hungarian students. (OECD 2011)

One year later after the digital PISA test, the frame curricula of 2013 was announced and published in Hungary (OFI 2012b), based on the Hungarian Core Curriculum (OFI 2012a). This document contains the curriculum content, output requirements and the required number of lessons per grade for each subject. In this national curriculum the number of classes of informatics and the years during which it was taught were drastically reduced, while the content and the requirements of the subject remained unchanged (Table 2). If there is a low number of classes, there is no negative correlation between students' digital reading scores and computer use at school, which will solve one of the worries of the PISA results, but will not produce any development in the students.

	grade											
	1	2	3	4	5	6	7	8	9	10	11	12
2009			0.5	0.5	0.5	1	1	1	1.5	1	1.5	1.5
2012						1	1	1	1	1		
2020			1	1	1	1	1	1	2	1	2	

Table 2. The indicated number of classes in each grade based on the frame curricula released in 2009, 2012 and 2020.

The frame curricula of 2012 (OFI 2012b) received a lot of criticism for its low number of classes and the fact that subject matter was not revisited. Teachers said the reduced number of lessons were not enough to teach the required content effectively. The educational process became further complicated, unverifiable, and uncontrollable by the vague descriptions of content in the frame curricula. Overly general expressions were frequent in these documents, leaving the correct terminology unattended to (Nagy & Csernoch 2018), as the examples in bold illustrate:

- basic management of **simple** application software,
- creation of a **short** drawing document,
- knowledge of the **basic** functions of the operating system,
- edit a document based on a **sample**

Consequently, ICT teachers interpreted the curriculum in different ways, depending on the extent and depth of their knowledge in the subject areas included in the frame curricula.

At the beginning of the academic year 2020/2021 in the newly introduced national curriculum the number classes of informatics was increased, almost to the number of the 2009 frame curricula (Table 2). In the most recent curriculum the content has been reorganized to some extent, but the content is still much more extensive than can be accomplished in the dedicated number of classes (OFI 2020). The number of topics related to programming has increased, and library informatics has been completely removed from the syllabus.

2. Theoretical background

ICT education needs a change in approach; the widespread and widely applied low-mathability, extremely inefficient surface approaches should be abandoned (Ben-Ari 1999; Panko 2008; Eusprig 2020) and replaced with more effective methods. A further problem is that most teachers teach the topics of informatics in isolation from each other. Furthermore, topics are strongly linked to one well-defined software programme. For example, text-management means Microsoft Word, and data-management Excel, and there is a strict focus on interface navigation by both teachers and students. They ignore document design, the criteria of correctly formatted and edited text, and the rules which should be applied to natural language texts. Instead of teaching the details of the software interface, the focus should be on the development of the students' digital competencies, digital literacy and computational thinking. Students should be taught how to master effective problem-solving strategies which would provide universal and long-lasting knowledge.

During my research, we have developed and applied teaching methodologies which were based on real-world problem solving. We have also designed a testing and evaluation system to examine the effectiveness of these methods, based on the national curricula and in accordance with previously administered competency tests. All our methods are based on the concept-based problem solving strategy introduced by György Pólya (1954), which has proven to be widely effective in various sciences, school subjects, and even in programming education within other fields of informatics. We placed great emphasis on activating the knowledge transfer elements within informatics. This approach allows us to build up connections with other sciences and subjects, and within informatics by connecting the different topics of computer sciences. The approach relies heavily on previously acquired knowledge – building and calling on schemas – which plays a key role in shaping long-term knowledge (Sweller 1988, Sweller et al. 2011, Kahneman 2011).

3. Webtable→datatable conversion for the file management topic

3.1 Measuring file management skills

In the Hungarian frame curricula one of the first topics is file management. Based on feedback and experience, this is the topic which is often neglected by teachers due to the digital natives' presumed knowledge. The importance of the topic was confirmed by the mini-competence test conducted in June 2008. This measurement focused on the students' computer and internet use habits and their ICT problem-solving competence (Nagy & Csernoch 2018). The main direction of my research is to map the knowledge first-year high school students bring from elementary school. Therefore, I was only interested in the results of the 7th (N=1562) and 9th (N=1643) grade students.

In the general section of the mini-competence test students filled in a self-assessment test on various ICT topics to evaluate their level of knowledge and indicated whether they had acquired knowledge of the topic at school (Figure 2). Considering file management, 79.06% of the students chose higher (3–5) proficiency levels, while 14.72% of them claimed that they did not learn file management in school. Those students whose did not learn at school were as confident in their knowledge as the others (did not learn level 4: 16.40%; level 5: 24.04%). This confidence seems to justify the idea that digital natives do not need any further education in file management.

G11. How would you rate your knowledge on the following topics? (0=not at all, 5=excellent)

	How familiar you are with the topics?						Learn in school?	
	Circle the correct number.						yes	no
file-management	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
word processing	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
spreadsheets	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
database management	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
algorithmizing, programming	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>
management of resources, credibility	0	1	2	3	4	5	<input type="checkbox"/>	<input type="checkbox"/>

Figure 2: A self-assessment test, part of the mini-competence test, determining the level of proficiency in ICT topics (Nagy 2018)

We tested the file management knowledge of first year high school students before we used the webtable→datatable conversion (WDC) method to teach the file management topic. This test was the pre-test of the efficiency measurement of the WDC method. 109 students took part in this test, all of them from the 9th grade. With this measurement our aim was to collect information on the knowledge of students arriving from elementary school. Because of the spiral education system of the national curriculum and its overly general expressions, officially, there is no difference between the knowledge of 7th and 8th grade students. Consequently, the students in the mini-competence test are at the same level as the students of the WDC measurement. Our further aim was to reveal the similarities, the correlations, and also the differences between the students' knowledge and their self-assessment. To compare the result of the two measurements, the WDC results were converted to a 0–5 scale. Based on the results obtained this way, the student's level of proficiency does not reach levels 4 and 5 (Figure 3) (Sebestyén 2020). The differences can be explained by the Dunning-Kruger effect, according to which the less proficient someone is in a field, the more confident they are of their knowledge of it, as they lack the knowledge needed to realistically assess their knowledge (Kruger & Dunning 1999).



Figure 3: The result of the mini-competence test (self-assessment) and the result of the pre-test of the WDC method (brought-in knowledge).

This comparison clearly indicates that digital natives also need ICT education, even for a seemingly simple topic such as file management. During the analyses of the knowledge students brought with them, we found that they have fragmented knowledge and cannot solve tasks that require the same knowledge element.

To teach file-management our research group applied the webtable→datatable conversion (WDC) method. During the teaching-learning periods the students used publicly available authentic contents from the Internet which matched their fields of interest and other school subjects. The students built the algorithm of the data handling process, and applied these steps to convert the webtable to a datatable (Csernoch & Dani 2017, Csernoch & Biró 2019). In the course of data processing, the students used data manipulation operations in text- and/or spreadsheet-management software. The idea came from the Sprego method (Csernoch 2014), where students work with authentic sources (Papp & Csernoch 2019) instead of the widely used decontextualized, randomly typed, meaningless sources. However, the table and table-like structures on the Internet cannot be used directly in the teaching-learning process in spreadsheet or database management environments. Consequently, these sources require pre-processing in advance to presenting them in data-management classes, whose major characteristic is handling various file types.

During the teaching period students learn and practice how they can convert webtables to normalized datatables. All through the problem solving process, students analyse the tables and data, and build algorithms to reach their goal, the normalized table. The conversation process is suitable for developing the students' computational thinking skills, which help them to choose the most effective tools and methods. The students acquire the terminology of the operating systems and file operations, which is needed so they can formulate the steps of the algorithm. During the conversion process several files of different types are created, which help students to clear up the misconceptions connected to file management – the data file can only be opened with the associated program; the file becomes unusable when the extension is changed, etc. The conversion requires creative solutions and ideas from the students, a process which can be guided by the teacher using the directed

inquiry-based method developed by Pólya (1954), reinvented recently as the coaching technique (Chmielewska & Gilányi 2019). One of the most important characteristics is that, in contrast with the tasks from the textbooks and ECDL (computer-cooking, without thinking) (Papp & Csernoch 2018, ECDL Foundation 2019, Dancsó & Korom 2013), we develop schemata which allow students to utilize slow and fast thinking effectively in problem-solving (Kahnemann 2011, Sweller 1988, Sweller et al. 2011).

The steps of the algorithm are unique in each conversion, so the students are required to present creative solutions in addition to the schema that forms the framework of the conversion. In each case, the students formulate their ideas together in a natural language, which makes the students more open and gives them the opportunity to think, to consider and transform each other's ideas.

4. The Error Recognition Model (ERM) for the text-management topic

4.1 Widespread methods

One of the IT application topics most emphasized in public education and the skill most applied in practice is text-management. In general, the term 'text-management' is falsely used interchangeably with Word, which is well presented in the official textbooks (Bánné et al. 2008; Bártfai 2011; Lakosné et al. 2019a, 2019b, 2019c; Dancsó & Korom 2013, Varga et al. 2020). In these textbooks, every topic starts with the presentation of the program environment and the details of the program window. Re-teaching this knowledge in each software environment is completely unnecessary and boring. This is because the program windows has the same structure, so the topic rather belongs to the teaching of operating systems than to the individual software applications. Furthermore, we must not forget that the goal is not to teach the software interface, but to solve problems effectively and build up long lasting knowledge which can be called upon to deal with novel problems. One further disadvantage of the interface-based approach, is that when newer versions arrive, one must start the learning process from the beginning. One further difficulty which must be faced during the word processing of natural

language texts is the lack of reliable spell- and grammar-checker applications. At the time of the present dissertation the effectiveness of the existing spell-checkers depends on the language, the correct layout and the author's language usage habits. So, the efficiency depends on the user's background knowledge. The most widely accepted approach is that a user relies primarily on the printed version or the printed image of the text (printed on paper or a digital interface), and if this is found acceptable, the user is satisfied.

In the text-management testing period, as with file management, we analysed the results of the mini competence test – in grades 7 and 8 –, and then compared them to the pre-test of the Error Recognition Model. In this case, we examined 3205 students, whose self-assessment results were compared with the text-management skills of 159 9th grade students.

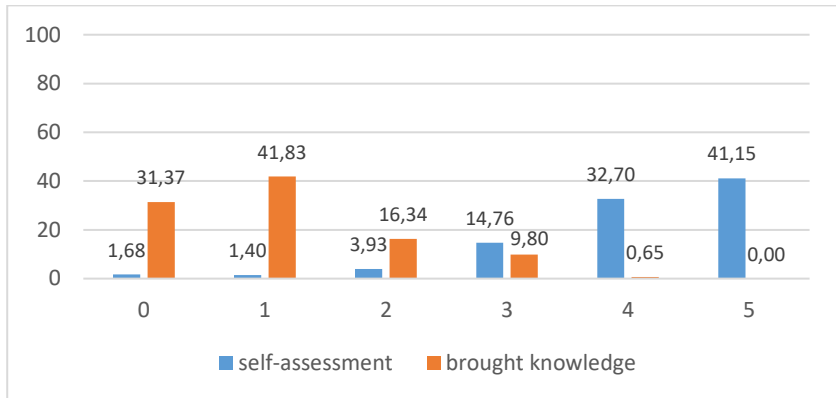


Figure 4: Comparison of the results of the mini-competency test (self-assessment) and the pre-test of ERM measurement .

There is no difference between the self-assessment values of the grade 7 and 8 students, even though there is a year of ICT learning between them. 40.85% and 37.80% of grade 7 and 8 students indicated the highest proficiency, respectively. The confidence of the grade 7 students clearly suggests the presence of the Dunning-Kruger effect (Kruger & Dunning, 1999). However, the difference between the self-assessment and the students' practical knowledge is inversely proportional (Figure 4). The discrepancy shows that

learners using traditional methods identify the topic by navigating the software interface.

Similarly to file management, students marked in the general section of the mini competency test whether they had learned text-management in school or not. 5.9% of the students answered that they had not studied this topic in school.

Another group of questions in the test dealt with the activities of ICT classes. Most of the students reported typing text (77.51%), followed by playing and free occupation (70.46%), while the third most popular response was typing tables (69.12%) (Nagy 2018). It is extremely frustrating that the most common activities include typing and playing in informatics classes. Typing cannot be the ultimate goal of an ICT class. Students type at different speeds and the results cannot be checked; moreover, it does not develop any of the skills and abilities required in teaching-learning informatics. Playing and free activities was the second most popular. One of my most difficult tasks as a practicing teacher was to make students understand that computer sciences and informatics is not the same as playing, but it is obviously still prevalent in many places. The results are in complete accordance with the results of a previous measurement (Sebestyén 2014) of 41 ICT teachers. 40% of these teachers allowed students to play during the lessons for varying periods of time (5–15 minutes). Nonetheless, they were dissatisfied with the frame curriculum (OFI 2012b), the number of classes, and the overloaded curriculum. It is also remarkable that only 16% of the teachers allowed playing in the other subjects they taught – typically maths, physics, and biology –, for a maximum of 5 minutes. This explains how it is possible that students have not learned text-management and why their proficiency in the topic is so low.

In text-management, our research group has also left the interface-based approaches behind. We focus on the appropriate combination of schema building that requires slow thinking, and based on these schemata, fast thinking is applied in solving novel problems (Kahneman 2011; Sweller et al. 2011). In the case of text-management, we use a modification of Pólya's concept-based problem-solving strategy (Pólya 1954), where the analysis, correction, and formatting of existing documents are carried out in advance of

creating new ones. This approach has proved effective in programming education (Gould 1975; Freiermuth et al. 2008; Bell & Newton 2013; Panko, 2013; Jerinic 2014; Gander 2014). Consequently, its adaption to natural language text-management seemed obvious.

In the ERM method, we distinguish three stages, depending on the extent to which we involve the use of the computer. The first one is the unplugged stage, during which students work without computers. Each student receives a printed document and uses a red pen to mark errors that can be identified in the printout. They indicate the location of the errors on the printed page and explain why they are incorrect. Effective error detection is aided by the categorization of errors based on rule systems that can be linked to natural language texts. Based on rules, the three categories of errors can be identified in printed texts: syntactic, semantic, and typographic (Csernoch 2019).

The second stage is semi-unplugged, where both the printed and the digital documents are required. Students open the digital version of the document and show the non-printable characters on the screen. Replacing the red pen with blue, students continue marking errors, but they are now looking for errors that can be identified in the digital document. Breaking-layout, formatting, and style errors can be recognized in a digital document.

The third and final stage is plugged-in, where students work with the computer. At this stage they correct the errors in the document they have recognized in the first two stages in order to obtain an error-free document which can be formatted effectively in the final steps of the procedure.

The advantage of this method is that the problem-solving process is divided into well distinguishable smaller processes (unplugged, semi-unplugged and plugged-in) to reduce the cognitive load which natural language texts require, and the distraction of computers. (Wolfram 2020).

5. Spreadsheet Lego (Sprego) for spreadsheet management

5.1 Widespread spreadsheet approaches

In contrast to word processors, spreadsheet applications use artificial language in their data processing. The advantage of artificial languages is that the program is able to check the expressions, and warn the user if a syntax

error is found. However, it is not syntax errors which are the most severe in spreadsheeting, as in other programming languages, but semantic errors. However, in most cases users do not check the correctness of the output debugging, and there is usually no discussion of the problem. Due to this negligent approach, as has been documented, 86% of spreadsheet documents are incorrect (Panko 2008), which can cause serious financial losses (Eusprig 2020).

In public education, teachers primarily apply interfaced-based, low-math-ability methods to teach spreadsheeting. Similar to text-management, the most common methods focus on the program interface, and navigation in the software environment. Tasks in the official textbooks begin mainly with the typing of a table of 4 to 5 meaningless records. Consequently, students do not understand why they should carry out any automated calculation if the solution is obvious without any effort. The high number of functions listed in the official coursebooks makes the problem more severe (Csernoch et al. 2014). The number of suggested functions is close to 100 because they are only suitable for solving one specific problem. The number of classes assigned to spreadsheeting, the high number of listed functions, the order of the arguments, the semantic aspects of the arguments, etc., all make it impossible to master the content.

An alternative solution to the ineffective traditional spreadsheeting and spreadsheet teaching is Sprego programming (Spreadsheet Lego). Sprego is a concept-based programming approach which uses the functional language of spreadsheet programs as the coding environment. Our previous measurements proved that the Sprego approaches is also suitable for teaching spreadsheet, data processing, and algorithmic and programming skills (Csapó et al. 2020, Csernoch et al. 2021, Sebestyén et al. 2019). Studies demonstrate that the Sprego can be used to teach spreadsheet significantly more effectively than the general interface centered approaches.

One of the great advantages of the Sprego methodology is that it supports the use of only twelve general purpose functions with which composite functions can be created if the problem so requires. Another advantage of Sprego programming is the simplicity of the language, so there is more room for

building up and discussing algorithms than in tradition imperative and object-oriented languages (Glevitzky & Csernoch 2021).

Understanding the problem-solving and algorithmic processes is aided by many unplugged tools (3D printed nested dolls, origami boats, commercially available barrel sets, countless other accessories) and semi-unplugged tools (2D and 3D training software) developed by our research team (Csapó & Sebestyén 2015, 2017, Sebestyén et al. 2018).

6. Hypotheses

[T1.] Schema and algorithm-based methodologies (WDC, ERM, Sprego) are significantly more effective than traditional, interface-based approaches.

For the methods developed by our research group, we designed and performed studies whose procedure was the same. Students are tested before the teaching in pre-tests, and after the learning process in post-tests to measure their development. To prove the effectiveness of the schema-based method, we used control groups studying with traditional interface methods widely accepted in public education. The involvement of control groups always required educators outside the research group. In some cases, the results of an individual study or group became invaluable due to the withdrawal of colleagues. This and the division of classes into two groups explains the smaller sample sizes. The comparison of pre- and post-tests was based on the results of students who participated in both tests, so self-improvement could be tracked.

WDC

The effectiveness of the WDC method was tested in the 2018/2019 academic year. The study included starting grade 9 students. The experimental group consisted of 30 students who mastered the topic of file management using the WDC method developed by the research group, while the control group consisted of 79 students who studied with the traditional methods (Sebestyén 2020). There was no significant difference ($p = 0.0607$) between the groups considering the brought-in knowledge in the pre-test (Figure 5).

At the end of the educational process, the study group achieved a significantly higher result compared to the control group ($p = 0.0000$). The result confirmed the effectiveness of the WDC method. Furthermore we cannot leave unremarked the fact that the control group showed no improvement at all in the post-test (37.88%) compared to the pre-test (38.02%) (Figure 5).

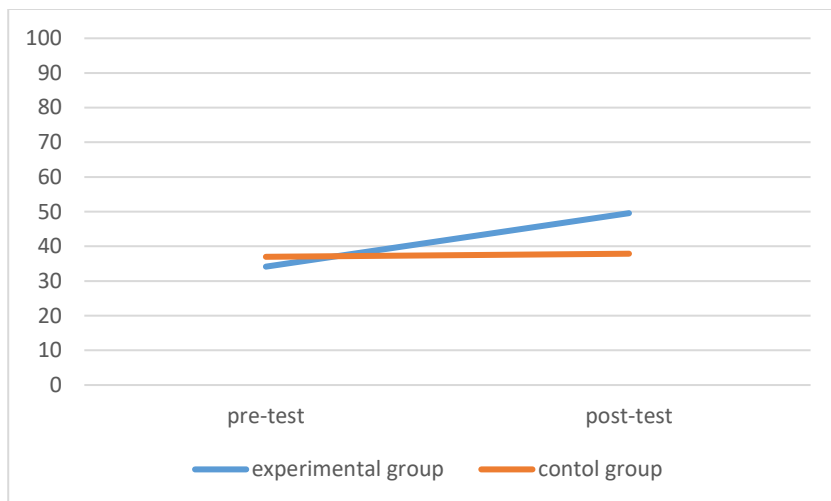


Figure 5: The students' result in the pre- and post-test of the WDC method.

ERM

The testing of the effectiveness of the ERM method was carried out in the 2017/2018 academic year. In addition to grade 9, the study included grade 7 students who also began their first year of high school in a six-grade high school course. According to the frame curricula, 9th grade students have already mastered the “basics” of word processing in previous years. A total of 153 students participated in the measurement (Table 3) (Sebestyén et al. 2021). The testing process followed the three stages of the teaching process (presented in the description of the method): unplugged, semi-unplugged, and plugged-in. There was no significant difference between the knowledge of students of the same grade in the pre-test. Based on the results of the paired tests, all groups improved significantly compared to the pre-test (Table 3). The pace of development of students in the same grade was the same. The 7V group developed the most – they learned with the ERM method –, even

though the students could not recognize the syntactic errors in the post test. The errors which can be recognized in the printed version can be classified into typographic, semantic and syntactic categories, the latter primarily meaning grammatical errors. The outstanding development of lower grade students suggests that the ERM method can be used effectively at this age. The error-recognition and problem-solving based schema-building structure of the ERM method ensures the development of long-term knowledge and produces more effective results than traditional approaches. This conclusion can be drawn from the low results of grade 9 students who studied text-management and word-processing for several years with interface-approach methods.

	experimental group				control group	
	grade 7 (7V)		grade 9 (9V)		grade 9 (9K)	
category	pre-test	post-test	pre-test	post-test	pre-test	post-test
syntactic	41.12%	19.75%	44.87%	27.02%	42.11%	36.27%
typography	0.00%	67.95%	0.00%	60.61%	0.00%	0.88%
layout	4.14%	41.73%	30.65%	49.32%	12.55%	36.58%
total	19.62%	39.68%	32.73%	45.23%	23.68%	34.74%

Table 3: The results of testing the effectiveness of the ERM method.

Sprego

The effectiveness of the Sprego methodology has already been proved in previous measurements (Csapó et al. 2019, Csernoch et al. 2021). As a result of our further joint work, we examined the long-lasting effect of the method, for which we designed a delayed post-test (Csapó et al. 2021). After the teaching period, approximately a year later, we measured students' knowledge again.

To administer the delayed post-test, those groups were selected who had studied spreadsheeting in the previous academic year. The experimental group consisted of 8th and 10th, while the control group 10th grade students, making 95 students altogether (Table 4). The groups studied spreadsheeting

in a different number of classes due to the different types of school programs (Table 4).

	experimental group				control group		
number of students	18	18	17	15	7	13	7
group name	e8_1	e8_2	e10_1	e10_2	c10_1	c10_2	c10_3
lessons / week	1	1	1	1	2	2	1
total number of lessons	8	8	6	6	12	12	6

Table 4: Number of the students participating in the Sprego delayed post-test and their number of classes covering spreadsheeting.

The aims of the delayed post-test was to examine how students analyze data, create algorithms, call on schemata and formulate functions.

	experimental group (%)					control group (%)				p
	e8_1	e8_2	e10_1	e10_2	all	c10_1	c10_2	c10_3	all	
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
T	38.73	49.38	55.08	50.37	48.20	14.52	28.54	7.20	19.37	0.0000

Table 5: The result of Sprego's long-term knowledge measurement (Csapó et al. 2021).

With the exception of one task, the experimental group achieved significantly higher results than the control group (Table 5). The e10_1 study group achieved the highest (55.08%), while the c10_3 group the weakest results. It is also worth paying attention to the results of lower grade students, where

the e8_2 group achieved similarly high results to the 10th grade students in the experimental group. The “weakest” experimental e8_1 group also had significantly better results than the two years older students from the control group (c10_1, c10_2, c10_3). Classes c10_1 and c10_2 of the control group are not able to approach the results of the experimental groups taught by the Sprego method, despite the higher number of lessons.

The testing of the effectiveness of the WDC and ERM method and the result of the delayed post-test of the Sprego method confirm the hypothesis. For all three methods, the experimental groups have significantly higher results than the control groups. In the case of younger students, it was found that their progression is more impressive than the older students who have studied for several years with the traditional method, both in text- and data-management.

[T2.] Knowledge transfer-based education within informatics is more effective than teaching independent topics.

During the acquisition of the problem-solving methods used by the research group, the students develop their computer thinking and algorithmic skills and abilities, which they can apply effectively in solving real-world problems. The effectiveness of the methods, in addition to the concept-based problem solving method and the contextualized teaching-learning environment, lies in the fact that it is based on both intra- and inter-subject relationships and knowledge transfer activities. While mastering the WDC method used in teaching file-management, students become familiar with various data manipulation operations that are typically performed in a text and spreadsheet environment. Data manipulation in a spreadsheet environment would prepare students for further studies in database management and programming.

The introduction of a new topic in the teaching-learning process is not a new chapter where all previous knowledge should be forgotten. Although there are different methods for each topic, each of them stimulates students to build and call on schemata and allows them to learn problem-solving strategies. Although the software interfaces and the range of tools continuously change, if students have software- and interface-independent knowledge they

will solve novel problems effectively, regardless of the immediate environment.

According to the frame curricula, after finishing a topic, students will not use the knowledge they have acquired until the topic is repeated in the following school years due to the spiral education system. One of the major disadvantages of this spiral structure is that 3–10 lessons are assigned to each topic, which is not enough to build up the knowledge required in the frame curricula.

On the contrary, our methods, based on building and calling on schemata and solving contextualized real-world problems, allow students to activate fast and slow thinking effectively.

The effectiveness of the methods used by the research team were tested separately, so we do not have statistical evidence of the knowledge transfer effects. These studies require further research over several years that goes far beyond the scope of the PhD. At present, based on our own experiences and on the opinions of colleagues and teachers involved in the research, the formulated thesis should not be rejected. The available data are suitable for preparing a longitudinal research to prove the thesis.

[T3.] Grade 9 high school students are either completely, or almost completely, unable to rely on the knowledge they have brought from primary school in the areas of file-, text-, and spreadsheet management.

The national ICT mini-competency test used by our research group was administered to 8880 students from grades 7–10 (Nagy 2018). Students filled in a self-assessment test on the given IT topics. It was found in both tests (WDC and ERM) that students' knowledge was burdened with misconceptions, and low results were achieved (WDC grade 9 32.34%, ERM grade 7: 20.29%; Grade 9: 28.62%). For tasks requiring the same knowledge items, students' responses were random, suggesting a lack of systematization of knowledge. These misunderstandings and results do not justify the level of proficiency indicated by the students themselves, but they do prove the thesis that students are either completely, or almost completely, unable to rely on the knowledge they bring from primary school.

[T4.] Students overestimate their ICT skills in self-assessment

The self-evaluation values of the topics of file-and text-management were compared to the results of the pre-test of the WDC and ERM which were completed by grade 9 high school students before studying the topics.

The comparisons for both the file-and text-management topics clearly demonstrate that students' abilities in solving knowledge-transfer based real-world problems were well below the values given in the self-assessment (Figure 3, Figure 4). In these cases, the Dunning-Kruger effect (Kruger & Dunning, 1999) can clearly be detected, as students claimed a high proficiency level with low background knowledge. These results show that self-assessment-based values do not provide a true picture of ICT competencies due to the low level of knowledge of students. Based on the results, this is thought-provoking, and further studies are needed to answer how the relationship between self-esteem and true knowledge develops at different ages and with the spreading of digital tools.

7. Future plans

Hypothesis [T2] could not be clearly proved, so I plan to design and prepare a further study that would allow us to measure the positive effect of knowledge-transfer.

To obtain more, and more reliable, results on the effectiveness of the deep-approach methods invented by our research group, both the number of groups and the students participating in the testing process should be increased. To complete this goal, the method should be available for schools and teachers; moreover, we are in great need of control groups for the comparative analyses. One of the most important tasks for the near future is to find teachers who are open to novelty and ready for change.

Considering the Sprego 2D educational program, I have designed several problems to illustrate and explain the algorithms and their evaluation process. However, the program has not been completed, so further amendments are required.

Seeing the effectiveness of the methods tested during my PhD studies, I am convinced that after obtaining the doctoral degree I would like to continue

the work with the research group to make changes in the approaches used in ICT education.

8. References

- Bánné Mászáros, A., Csintalan, T. & Lakosné Makár, E. (2008). Informatics 5. In Hungarian: Informatika 5. Nemzeti Tankönyvkiadó, Budapest.
- Bártfai, B. (2011). Computer use for everyone. In Hungarian: Számítógéphasználat mindenkinek. BBS-INFO Könyvkiadó és Informatikai Kft., Budapest.
- Bell, T. & Newton, H. (2013). Unplugging Computer Science. In Improving Computer Science Education. (Eds.) Djordje M. Kadijevich, Charoula Angeli, and Carsten Schulte. Routledge. pp. 66-81.
- Ben-Ari, M. (1999). Bricolage Forever!, PPIG 1999. 11th Annual Workshop. 5–7 January 1999, Computer-Based Learning Unit, University of Leeds, UK, 1999, pp. 1-5. Retrieved 2020.10.10. from <https://ppig.org/files/1999-PPIG-11th-benari.pdf>
- Chmielewska, K. & Gilányi, A. (2019). Computer Assisted Activating Methods in Education. 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Naples, Italy, 2019, pp. 241-246.
- 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. Retrieved 2021.05.10. from <https://link.springer.com/content/pdf/10.1007/s10639-019-10024-2.pdf>
- Csapó, G. & Sebestyén, K. (2015). Application for the Sprego method. In Hungarian: Oktatóprogram a Sprego táblázatkezelő módszerhez. INFODIDACT 2015: Informatika Szakmódszertani Konferencia. Retrieved 2020.02.12. from <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. Retrieved 2020.06.12. from http://www.tojet.net/special/2017_10_1.pdf
- 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. (2014). Programming with spreadsheet functions. In Hungarian: Programozás táblázatkezelői függvényekkel. Műszaki Tankönyvkiadó, Budapest.

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, 2019, pp.147-152.

Csernoch, M. & Biró, P. (2015) Wasting Human and Computer Resources. International Journal of Social Education Economics and Management Engineering, 9(2), pp. 555-563.

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. Abari, K. & Máth, J. (2014). Programming-oriented spreadsheet functions. In Hungarian: Programozásorientált táblázatkezelői függvények. XIV. Országos Neveléstudományi Konferencia: Oktatás és nevelés – gyakorlat és tudomány : Tartalmi összefoglalók. Debreceni Egyetem Neveléstudományok Intézete (2014), pp. 463-463.

Csernoch, M.; Biró, P. & Máth, J. (2021). Developing computational thinking skills with algorithm-driven spreadsheeting. IEEE Access. 9. pp. 153943-153959.

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.

Dancsó, T. & Korom, P. (2013). Informatics 9-10. In Hungarian: Informatika 9-10. – A gimnáziumok számára. Nemzedékek Tudása Tankönyvkiadó. Budapest.

ECDL Foundation (2019). ECDL Tasks. In Hungarian: ECDL vizsgapéldatár. Számítógépes-alapismeretek. Syllabus 1.0. Neumann János Számítógép-tudományi Társaság.

Eusprig (2020). Horror Stories. Retrieved: 2020.10.10. from <http://eusprig.org/horror-stories.htm>

Freiermuth, K., Hromkovic, J. & Steffen, B. (2008). Creating and Testing Textbooks for Secondary Schools. In: Mittermeir R.T., Sysło M.M. (eds) Informatics Education - Supporting Computational Thinking. ISSEP 2008. Lecture Notes in Computer Science, vol 5090. Springer, Berlin, Heidelberg. pp. 1-13.

- Gander, W. (2014). Informatics and General Education. In: Gülbahar Y., Karataş E. (eds) Informatics in Schools. Teaching and Learning Perspectives. ISSEP 2014. Lecture Notes in Computer Science, vol 8730. Springer, Cham. pp. 1-49.
- Glevitzky, K. & Csernoch, M. (2021). Solving advanced ICT graduation programming tasks with Sprego method. In Hungarian: Az emelt szintű informatika érettségi programozás feladatainak megoldása Spregoval. Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2021.
- Gould, J. (1975). Some psychological evidence on how people debug computer programs. *International Journal of Man-Machine Studies*, (7) 1, pp. 151-182.
- 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*, (5) 6. pp. 53-59.
- 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 - Teaching and Learning in the New Digital Landscape. SAGE Publications. Thousand Oaks, United States.
- Kahneman, D. (2011). Thinking, Fast and Slow. New York: Farrar, Straus and Giroux
- 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), 75-6.
- Kirschner, P. & De Bruyckere, P. (2017). The myths of the digital native and the multitasker, *Teaching and Teacher Education*. 67 (2017), pp. 135–142.
- 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.
- Lakosné Makár, E., Rajk, Á., Regele, Gy. & Ridzi, G. (2019a). Informatics 6. In Hungarian: Informatika 6., OFI. Retrieved 2020.10.01. from https://www.tan-konyvkatalogus.hu/pdf/NT-11682__teljes.pdf

- Lakosné Makár, E., Rajk, Á., Regele, Gy. & Ridzi, G. (2019b). Informatics 7.. In Hungarian: Informatika 7., OFI. Retrieved 2020.10.01. from https://www.tankonyvkatalogus.hu/pdf/NT-11782__teljes.pdf
- Lakosné Makár, E., Rajk, Á., Regele, Gy. & Ridzi, G. (2019c). Informatics 8.. In Hungarian: Informatika 8., OFI. Retrieved 2020.10.01. from https://www.tankonyvkatalogus.hu/pdf/NT-11882__teljes.pdf
- Nagy, T. K. (2018). . Analysis of the grade 7-10 students' computer transfer thinking. In Hungarian: 7–10. évfolyamos tanulók tudástranszfer alapú számítógépes gondolkodásának elemzése reprezentatív felmérés alapján. Debreceni Egyetem Informatikai Kar Tudományos Diákköri Konferencia 2018. Retrieved 2020.05.10. from <http://hdl.handle.net/2437/277191>
- Nagy, T. K. & Csernoch, M. (2018). The paradox of the hungarian frame curricula in informatics. The Turkish Online Journal of Educational Technology, INTE 2018. pp. 783-795. Retrieved 2020.09.12. from http://www.tojet.net/special/2018_12_3.pdf
- OECD (2011). PISA 2009 Results: Student On Line: digital technologies and performance (Volume VI). Retrieved 2020.02.10. from <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. Retrieved 2020.02.10. from <https://www.oecd.org/pisa/pisaproducts/PISA-2012-technical-report-final.pdf>
- OFI (2012a). Hungarian Core Curriculum. In Hungarian: 110/2012. (VI. 4.) Korm. rendelete a Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról. Retrieved 2020.12.10. from http://ofi.hu/sites/default/files/attachments/mk_nat_20121.pdf
- OFI (2012b). Hungarian Curriculum Framework. In Hungarian: 51/2012. (XII. 21.) számú EMMI rendelet – a kerettantervek kiadásának és jóváhagyásának rendjéről. Retrieved 2020.11.19. from https://www.oktatas.hu/koznevelés/kerettantervek/2012_nat
- OFI (2020). Hungarian Core Curriculum. In Hungarian: 5/2020. (I.31.) Korm. rendelet: .A Nemzeti alaptanterv kiadásáról, bevezetéséről és alkalmazásáról szóló 110/2012. (VI.4.) Korm. rendelet módosításáról. Retrieved 2020.11.15. from

<https://magyarkozlony.hu/dokumentumok/3288b6548a740b9c8daf918a399a0bed1985db0f/megtekintes>

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. (2013). The Cognitive Science of Spreadsheet Errors: Why Thinking is Bad. *Proceedings of the 46th Hawaii International Conference on System Sciences*, January 7-10, 2013, Maui, Hawaii.

Panko, R. R. & Aurigemma, S. (2010). Revising the Panko-Halverson taxonomy of spreadsheet errors. *Decis. Support Syst.* 49, 2., pp. 235-244.

Papp, P. & Csernoch, M. (2018). Is spreadsheeting equal to problem solving. In Hungarian: A táblázatkezelés is problémamegoldás? *Infodidact 2018*. Retrieved 2020.11.10. from <https://people.inf.elte.hu/szlavi/InfoDidact18/Manuscripts/PPCsM.pdf>

Pólya, G. (1954). *How To Solve It. A New Aspect of Mathematical Method*. Princeton University Press, Princeton, New Jersey.

Prensky, M. (2001). *Digital Natives, Digital Immigrants, From On the Horizon* (MCB University Press, Vol. 9 No. 5, October 2001)

Sebestyén, K. (2014). You can also be an IT teacher. In Hungarian: Te is lehetsz informatikatanár (?). *INFODIDACT 2014*. Retrieved 2020.10.01. from <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 Rep-representations. *Turkish Online Journal of Educational Technology, INTE 2018* November, pp. 1081-1091 Retrieved 2020.05.05. from 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. Retrieved 2020.06.08. from 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*. Accepted.
- Szabados, S. (2009). Digital natives. In Hungarian: Digitális bennszülöttek. *Oktatás-informatika I. évfolyam, 2009. 1. szám*, pp. 19-23.
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning, *Cognitive Science*, 12, pp. 257-285.
- Sweller, J., Ayres, P. & Kalyuga, S. (2011). *Cognitive Load Theory*. Springer.
- U. S. Department of Education (2018). The Condition of Education 2018. NCES 2018-144. IES. Retrieved 2020.10.12. from <https://nces.ed.gov/surveys/icils/icils2018/pdf/2019164.pdf>
- Varga, P., Jencsiné Horváth, K., Reményi, Z., Farkas, Cs., Takács, I., Siegler, G. & Abonyi-Tóth, A. (2020). Digital Culture 9.. In Hungarian: Digitális kultúra 9. Oktatási Hivatal, Budapest. Retrieved 2021.02.18. from https://www.tankonyvkatologus.hu/pdf/OH-DIG09TA__teljes.pdf
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), pp 33-35.
- Wolfram, C. (2020). *The Math(s) Fix: An Education Blueprint for the AI Age*. Wolfram Media, Inc.



Registry number: DEENK/484/2021.PL
Subject: PhD Publication List

Candidate: Katalin Sebestyén
Doctoral School: Doctoral School of Informatics
MTMT ID: 10054523

List of publications related to the dissertation

Foreign language scientific articles in Hungarian journals (2)

1. **Sebestyén, K.**, Csapó, G., Csermoch, M., Aradi, B.: Error Recognition Model: End-user Text Management.
Acta Polytech. Hung. "Accepted by Publisher" (-), 1-20, 2021. ISSN: 1785-8860.
IF: 1.806 (2020)
2. **Sebestyén, K.**, Csapó, G., Csermoch, M., Abari, K.: Measuring the algorithmic skills of students working with low- and high-mathability programming approaches.
Acta Polytech. Hung. "Accepted by Publisher" (-), 1-20, 2021. ISSN: 1785-8860.
IF: 1.806 (2020)

Foreign language scientific articles in international journals (1)

3. Csapó, G., **Sebestyén, K.**, Csermoch, M., Abari, K.: Case study: developing long-term knowledge with Sprego.
Educ. Inf. Technol. 26 (1), 965-982, 2021. ISSN: 1360-2357.
DOI: <http://dx.doi.org/10.1007/s10639-020-10295-0>
IF: 2.917 (2020)

Hungarian conference proceedings (1)

4. **Sebestyén, K.**: Te is lehetsz informatikatanár (?).
In: INFODIDACT 2014. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika Alapítvány, Zamárdi, 1-7, 2014. ISBN: 9789631206272

Foreign language conference proceedings (4)

5. **Sebestyén, K.**: Students' Knowledge in File Management.
In: Proceedings of the 11th International Conference on Applied Informatics (ICA I 2020), Ed.: by Gergely Kovásznai, István Fazekas, Tibor Tómacs, CEUR, Eger, 296-308, 2020, (CEUR Workshop Proceedings, ISSN 1613-0073 ; 2650.)
6. **Sebestyén, K.**, Csapó, G., Csermoch, M.: Introduction to Algorithmic Based Data Management in Spreadsheet Environment.
Turk. Online J. Educ. Technol. 2019 (Spec.), 178-185, 2019. EISSN: 1303-6521.





7. Biró, P., Csernoch, M., **Sebestyén, K.**, Gombos, E.: Algorithmic Skills Vs. Time Spent On Computers And Mobile Devices.
Turk. Online J. Educ. Technol. 1 (Spec.), 133-141, 2018. ISSN: 2146-7242.
8. **Sebestyén, K.**, Csapó, G., Csernoch, M.: Visualising Sprego Inequality Problems With 2D Representations.
Turk. Online J. Educ. Technol. Special Issue for INTE-ITICAM-IDEA 2018, 888-898, 2018. EISSN: 1303-6521.

List of other publications

Hungarian conference proceedings (1)

9. Csapó, G., **Sebestyén, K.**: Oktatóprogram a Sprego táblázatkezelő módszerhez.
In: INFODIDACT 2015. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika Alapítvány, Zamárdi, 1-18, 2015. ISBN: 9789631238921

Foreign language conference proceedings (1)

10. Csapó, G., **Sebestyén, K.**: Educational software for the Sprego method.
Turk. Online J. Educ. Technol. Special Issue for INTE 2017, October, 986-999, 2017. EISSN: 2146-7242.

Hungarian abstracts (1)

11. Csapó, G., **Sebestyén, K.**: Bevezetés az algoritmus alapú adatkezelésbe táblázatkezelő környezetben: Digitális Témahét tapasztalatai.
In: ADA 2019 : Rövid absztraktok, Debrecen Egyetemi Kiadó, Debrecen, 8, 2019.

Total IF of journals (all publications): **6,529**

Total IF of journals (publications related to the dissertation): **6,529**

The Candidate's publication data submitted to the iDEa Tudóstér have been validated by DEENikon on the basis of the Journal Citation Report (Impact Factor) database.

04 November, 2021

