

Doktori (PhD) értekezés tézisei

**Interaktív, többérzékszerves elektronikus
oktatási környezet a számítógépes
gondolkodás fejlesztéséhez**

Osztían Erika

Témavezető: Dr. Fazekas Gábor



DEBRECENI EGYETEM

Matematika- és Számítástudományok Doktori Iskola

Debrecen, 2021

BEVEZETŐ

Az oktatásnak igazodnia kell a XXI. századi igényekhez. Nekünk, tanároknak, fel kell készítenünk a generációkat a mindennapi életünkben már jelenlevő és egyre növekvő informatizálásnak a kihívásaira. A végzős diákoknak bátran kell szembenézniük a munkaerőpiac követelményeivel és tudniuk kell teljesíteni mindezeket. E gondolatok következményeként, fontos lenne számukra, hogy értsék az informatikai és az ehhez kapcsolódó készségek alapfogalmait. Meg kell ismerkedniük a számítógépes gondolkodás fogalmával, amely napjainkban egy alapvető készséggé fejlődött az olvasás, írás és aritmetika mellett, és amelyet a modern társadalmunkban mindenkinek birtokolnia kellene. (Denning, 2009). Ennek elérése érdekében nagyon fontos szerepet játszik az oktatási környezet.

A tanulók számítógépes gondolkodásának fejlesztésében a számítógépes algoritmusoknak kiemelt szerepük van. Az algoritmusok elvont folyamatok, azért annak érdekében, hogy mindenki megtalálja a helyes utat, amely elvezet ennek a fogalomnak a megértéséhez, egy megoldás az lehet, hogy ezeket olyan kontextusban mutassuk be, amely bárki számára vonzó tud lenni. A vizualizáció egy kritikusan fontos eszköz lehet a hatékony oktatásukhoz.

Az AlgoRhythmics kutatócsoportunk, amely a Sapientia Erdélyi Magyar Tudományegyetem keretében működik, a megoldást az informatika és a tánc párosításában látja.

Művészegyüttesekkel karöltve, tánckoreográfiákat dolgoztunk ki, amelyek alapvető számítógépes algoritmusokat illusztrálnak.

Az AlgoRhythmics környezet, amely az algoritmus táncok és számítógépes animációkat összesíti, a tanulók algoritmikus gondolkodásának fejlesztése céljából született meg. Arra építkeztünk, hogy a nem hagyományos oktatási stratégiák bevezetésével, a diákok fogékonyabbá válhatnak a tanulást illetően (Kátai et al., 2008).

Egy tánckoreográfia vagy egy dallam esetén, a művészi elemek által hozzáadott érték szárnyakat adhat a tanuláshoz. Egy gondosan megtervezett absztrakt vizualizáció segíthet elvonatkoztatni attól, ami a vizsgálat szempontjából lényegtelen, és a tanulmányozott algoritmus kulcsműveleteire fókuszálni. Ez a fajta tanulás a többbérzekszeres tanulás elvére épül. Comenius elvét szem előtt tartva, az általunk fejlesztett környezetet is így építettük meg. Az építkezés, lépésről-lépésre, érzékszervről-érzékszervre, modulárisan történt. Mivel az aktív tanulás elve is figyelembe volt véve, a kezdeti ötlet egy többbérzekszeres interaktív rendszerré fejlődött.

A kutatócsoport 2007-ben tette le alapköveit. Ahhoz, hogy a gondosan kidolgozott algoritmus koreográfiák művészeti szempontból is értékesek legyenek, együttműködést születtetett a Maros művészegyüttessel (néptáncosokkal), akik segítségével a rendezési algoritmusokat illusztráló tánckoreográfiákat művészileg is színvonalassá tették. Ezen kollaboráció gyümölcseként született meg 2011-ben 6 rendezési algoritmus videóra vett tánckoreográfia illusztrációja. Ami az alkalmazott tánc típusokat illeti, az Erdélyben honos etnikumok néptáncai, magyar, székely, román, cigány, szász, csángó, tették színessé a koreográfiákat.

A YouTube-on (www.youtube.com/user/AlgoRhythmics) közzétett videók nézettsége, valamint az itt megosztott biztató megjegyzések újabb szárnyakat adtak a környezet továbbfejlesztésére.

Az AlgoRhythmics módszer kiteljesedését azok az absztrakt animációk jelentették, amelyeknek vitathatatlan előnye, hogy elvonatkoztatnak a lényegtelen részletektől, és segítenek az algoritmusok kulcselemeire összpontosítani. Így született meg az AlgoRhythmics webes környezetet, amelyben a tánckoreográfia ábrázolások mellé egy-egy absztrakt animációt társítottunk. Ezen új model egy további erőssége az volt, hogy interaktív üzemmódban is működött.

Annak köszönhetően, hogy az AlgoRhythmics alkalmazás egyedi módon kombinálja a tudományt a művészetekkel, ezen eszköz hatékonyságát reál és humán szakos hallgatókon is alkalmazták. Az akkor született eredmények nagyon biztatóak voltak (Kátai & Adorjáni, 2014; Kátai, 2015).

A projektet 2016-tól továbbfejlesztve, az algoritmus-kollekció a keresési stratégiákkal gazdagodott, az algoritmusok megjelenítése kiegészült kétdimenziós ábrázolásokkal, melyeket új táncstílusok: a flamenco és balett kísért. Teljesen újratervezve a webes alkalmazást, olyan modulokkal bővült, amelyek elvezették a diákokat az algoritmusok megértésétől ezek implementálásáig.

AZ ÉRTEKEZÉS ÚJ TUDOMÁNYOS EREDMÉNYEI

1. Sematikus versus emberi mozgást alkalmazó realisztikus algoritmusvizualizáció

TÉZISEK

[T1.1] Azok a realisztikus algoritmusvizualizációk (mint például az AlgoRhythmics tánckoreográfiák), amelyek emberi mozgás révén illusztrálják az algoritmikai tartalmat, ugyanolyan hatékonyak, sőt hatékonyabbak is tudnak lenni, mint a megfelelő sematikus ábrázolások (absztrakt számítógépes animációk).

[T1.2] Könnyebb különbséget tenni a megtekintett algoritmusok stratégiái között, amennyiben ezeket sajátos realisztikusvizualizációk ábrázolták, mintha sematikus ábrázolások lettek volna alkalmazva.

[T1.3] Az olyan realisztikus algoritmusvizualizációk esetében, amelyek emberi mozgást alkalmaznak, főleg azok a figyelemelterelő elemek kerülendők, amelyek árnyékot vetnek azokra a mozgásokra, amelyek az algoritmus kulcsműveleteit illusztrálják (kevésbé jelentenek gondot az esetleges dekorációs figyelemelterelők).

Mivel az algoritmusok természetükben dinamikusak, ezért a legkézenfekvőbb, hogy dinamikus vizualizációkat alkalmazzunk az illusztrálásukhoz, mint amilyenek az animációk és a videók.

Általában egy dinamikus vizualizáció lehet: sematikus vagy realisztikus. (Nugteren et al., 2014). Ebben a kutatásban arra fókuszáltunk, hogy felmérjük e két ábrázolási mód egymáshoz viszonyított értékét (a vizsgált tartalom megértéséhez való hozzájárulásuk tekintetében) az algoritmus vizualizáció témakörben. Mivel az algoritmusok elvont folyamatok, megszokottá vált, hogy sematikus ábrázolásokkal, azaz absztrakt animációk által illusztrálják működésüket. Másfelől viszont születtek olyan vizualizációk is, főként az unplugged informatika oktatás keretében, amelyek közelebb állnak a realisztikus ábrázoláshoz. Egy ilyen példa az AlgoRhythmics (2020) környezet is, amely realisztikus tánckoreográfiákkal illusztrál számítógépes algoritmusokat. Ezeknek egy sajátossága, hogy emberi mozgást is tartalmaznak.

Figyelmmel követtük a vizualizációk reprezentációs, illetve dekorációs figyelemelterelő elemeit. Reprezentációs elemnek számít az, amelyik a tanulás tárgyát képező jelenség ábrázolására szolgál. Ezzel szembe a dekorációs elemek elsődleges szerepe az, hogy motivációt gerjesszenek. Különösen érdekeltünk abban, hogy mennyire érinti ez a

kétfajta figyelemelterelő elem az AlgoRhythmic videók hatékonyságát. Ezért a kísérlethez két algoritmust választottunk: egy olyat, amely esetében a tánckoreográfia csak dekorációs természetű potenciális-figyelemelterelőket tartalmaz (kiválasztó rendezés), és egy olyat, amely esetében jelen vannak a reprezentációs jellegű potenciális-figyelemelterelők is (buborékredezés).

A kutatás

A kísérletet háromszakaszosra terveztük: előteszt, tanulási szakasz, utóteszt. A kísérletre a 2019/20-as tanév elején került sor, az első héten az előtesztre, a másodikon a tanulási szakaszra és az utótesztre. Az előteszten 137 diák, az utóteszten 116 olyan hallgató vett részt. A 116 résztvevőnek fele (58 diák) volt kontroll csoportbeli, és a másik fele (58 diák) tartozott a kísérleti csoporthoz. Azokat a hallgatókat tekintettük kontroll csoportnak, akik (s)ematikus ábrázolással (absztrakt számítógépes animációkkal) tekinthették majd meg a kiválasztott algoritmusokat (S-csoport). Mivel a környezet újszerűsége főleg a tánckoreográfiákban rejlik, ezért azok a hallgatók képezték a kísérleti csoportot, akiknél a (r)eálisztikus ábrázolásokat alkalmaztuk (R-csoport).

Eredmények és kiértékelés

Az utóteszt eredményeinek az összehasonlítására az ANCOVA tesztet használtuk. A független változó újra a tanmenet típusa volt, a függő változó pedig az utóteszt kérdésekre adott válaszok pontszámai. Az előteszt eredményeket, mint kovariáns használtuk. Az elemzés azt mutatta, hogy nincs szignifikáns különbség a két csoport között a buborékredezésre vonatkozóan ($F(1,113)=0,78$, $p=0,37>0,05$). A kiválasztó rendezés esetében viszont a kísérleti csoport (R-csoport) szignifikánsan jobban teljesített, mint a kontroll csoport (S-csoport) ($F(1,113)=9,52$, $p=0,002<0,05$; partial $\eta^2 = 0,07$) (1. táblázat). Az ANCOVA teszt is azt mutatta, hogy ezt az eredményt nem befolyásolta az előtesztnél tapasztalt minimális (nem szignifikáns) különbség.

1. Előteszt		2. Utóteszt	
		3. Buborékredezés	4. Kiválasztó rendezés
5. R-csoport	6. 68%	7. 65%	8. 58%

9. S-csoport	10. 69%	11. 68%	12. 45%
--------------	---------	---------	---------

1. táblázat: Elő- és utóteszt eredmények a két csoport szignifikancia különbségének mérésére

A bemutatott kutatás eredményeiből arra következtethetünk, hogy amennyiben egy realisztikus vizualizáció emberi mozgást alkalmaz, ugyanolyan hatékony tud lenni, sőt akár hatékonyabb is, mint a megfelelő sematikus ábrázolás. Ez azért fontos következtetés, mert korábbi kutatások inkább a sematikus vizualizációk hatékonyságát hangsúlyozták, merthogy kevesebb figyelemelterelő elemet tartalmaznak (T1.1 tézis).

Előző eredményekkel összhangban újra megerősítést nyert, hogy a realisztikus ábrázolásokra könnyebb visszaemlékezni és több rokon ábrázolás között könnyebb különbséget tenni. A jelen kutatás kiterjesztette ezt az eredményt az algoritmus vizualizációk területére is (T1.2 tézis).

Definiáltuk, dinamikus vizualizációkra vonatkozóan, a dekorációs és reprezentációs figyelemelterelők fogalmát. Az eredmények arra mutatnak, hogy főleg a reprezentációs figyelemelterelő elemek kerülendők, az a jelenség amikor „árnyék vetődik” azokra a mozgásokra, amelyek az algoritmikai tartalmat illusztrálják (T1.3).

Mindezen következtetések, amelyeket tézisenként is megfogalmaztunk, hasznos információval szolgálhatnak mindazoknak, akik algoritmus vizualizációs környezeteket terveznek és használnak az oktatásban.

Megjelent publikációk

Idegen nyelvű konferencia közlemény [8]

Magyar nyelvű absztrakt kiadvány [1]

2. Sematikus és realisztikus algoritmus vizualizációk ismételt megtekintése, illetve ezek kombinálása

TÉZISEK

[T2.1] Amennyiben ismételt megtekintést alkalmazunk, az emberi mozgás effektust hasznosító realisztikus algoritmusvizualizációk (mint például az AlgoRythmics videók) hatékonyabbak, mint a megfelelő sematikus ábrázolások (például absztrakt számítógépes animációk).

[T2.2] Amennyiben sematikus és realisztikus algoritmusvizualizációt is alkalmazunk, előnyösebb lehet, ha előbb a sematikus, majd a realisztikus kerül megtekintésre. Mivel a realisztikus vizualizációk könnyebben felidézhetők, mint a sematikusak, ezért ismételt megtekintés, illetve több ábrázolás kombinálása esetén előnyös, ha a tanmenet realisztikus ábrázolásra végződik.

A kutatás

A dinamikus vizualizációkkal való tanulás nem minden esetben hatékonyabb, mint a statikus képek alkalmazása. Mindez a videók/animációk tranziens jellegéből adódhat. Kézenfekvő arra gondolni, hogy elősegítheti a dinamikus vizualizációval való tanulási folyamatot, ha a diákoknak lehetőségük van többször is megtekinteni a vizualizációt. Ennek következményeként, amíg az előző kutatásunkban a sematikus, valamint a realisztikus algoritmus vizualizációk egymáshoz viszonyított értékeit vizsgáltuk, ebben a kutatásban arra összpontosítottunk, hogy milyen előny származhat ezek többszöri megtekintéséből („sematikus+sematikus”, „realisztikus+realisztikus”), illetve a kombinálásukból („sematikus+realisztikus”, „realisztikus+sematikus”) (1. ábra).

Erre a kísérletre a 2018-19 tanév elején került sor, és három szakaszban vizsgáltuk a 84 (14% lány) hallgatót előzetes programozói ismerettel rendelkező hallgatók válaszait (1, 2 vagy 4 évig tanultak informatikát a középiskolában)

Ahhoz, hogy négy, statisztikailag egyenértékű csoportot alakítsunk ki, rendeztük a résztvevőket az előteszt eredményeik alapján, majd annak függvényében soroltuk őket csoportokba, hogy a sorszámuk 4-gyel való osztási maradéka 0, 1, 2 vagy 3 volt. A csoportok az alábbi tanmenetet követve sajátították el a Shell-sort rendezési algoritmust (ez az algoritmus nem szerepelt az addigi tantervükben):

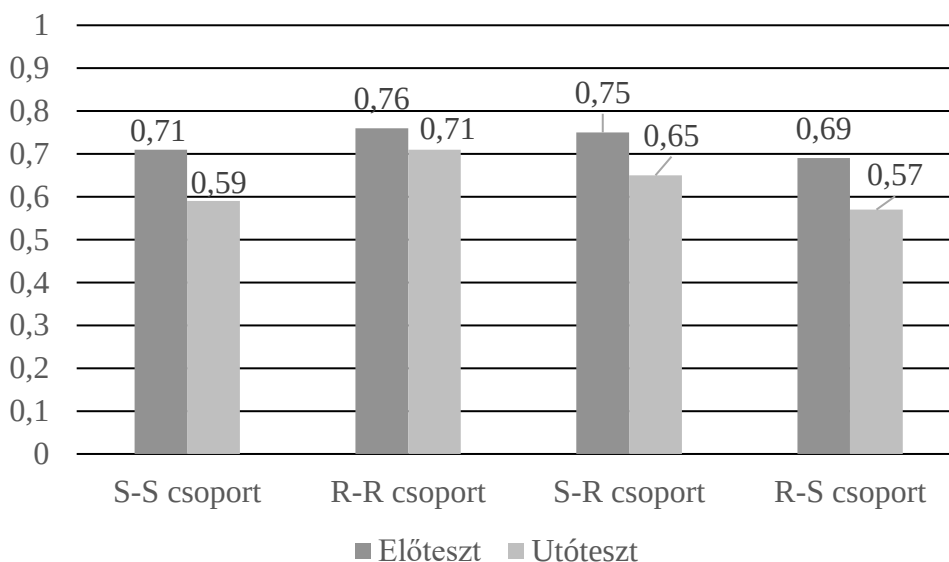
Animáció	Videó	Videó	Animáció
+	+	+	+
Animáció	Videó	Animáció	Videó



1. ábra: Statikus vs. dinamikus ábrázolásmód

Eredmények és kiértékelés

Az utóteszt eredmények arra mutatnak, hogy realisztikus ábrázolások esetén kiváltképpen hasznos lehet egy második megtekintés. (2. ábra)



2. ábra: Az első kísérlet eredményeinek ábrázolása

Az a tény, hogy azok a diákok, akik kétszer tekintették meg a realisztikus ábrázolást szignifikánsan jobban teljesítettek, mint azok, akik a sematikus látták kétszer, nem csak arra enged következtetni, hogy az emberi mozgást alkalmazó realisztikus ábrázolások hatékonyabb eszközök, mint a megfelelő sematikusak, hanem arra is, hogy a kétszeri megtekintésnek több hozadéka volt az R-R csoportnál, mint az S-S csoportnál (T2.1 tézis).

A jelen vizsgálat nyomán, algoritmus vizualizációs környezetben is beigazolódott, hogy annak is meglehet az előnye (illetve hátránya), ha ugyanannak az algoritmusnak két izomorf vizualizációját alkalmazzuk (S-R és R-S csoportok), és annak is, ha ugyanazt a vizualizációt kétszer tekintik meg a diákok (S-S és R-R csoportok) (T2.2 tézis).

Bár a különbség nem volt szignifikáns (S-R versus R-S csoport), a mi eredményeink is azt sugallják, hogy amennyiben kétféle ábrázolást is alkalmazunk, előnyös lehet először a sematikus (az egyszerűbbet, amelyik jobban igazodik a diákok előzetes ismereteikhez) lejátszani, és azután a realisztikus (az összetettebbet, az újszerűbbet) (T2.3 tézis).

Végezetül újra igazolást nyert, hogy a sajátos jegyeket tartalmazó realisztikus ábrázolásokra könnyebb visszaemlékezni, és ez lehet az oka annak, hogy azok a csoportok, amelyek másodikként a realisztikus vizualizációt látták (R-R és S-R), szignifikánsan jobban teljesítettek, mint a másik két csoport (S-S és R-S) (T2.4 tézis).

Megjelent publikációk

Idegen nyelvű konferencia közlemény [6]

3. Az interaktivitás szerepe online algoritmus-vizualizációs környezetekben

TÉZISEK

[T3.1] A előzetes programozási ismeretekkel nem rendelkező (kezdők – 0 év programozás) és kevés programozási tapasztalattal rendelkező (középhaladók – 1 vagy 2 év programozás) diákok javára válhat az algoritmus-vizualizáció interaktív bemutatása. A haladók (3 vagy 4 év programozás) otthonosan mozognak bármely interaktivitási szinttel rendelkező tanfolyam esetén.

[T3.2] Nincs szignifikáns különbség a különböző interaktivitási szintekkel tanuló fiúk és lányok eredményei között

[T3.3] Ha az interaktivitásból adódóan nagyon feldarabolódik az algoritmusvizualizációs folyamat, akkor ez megnehezítheti az algoritmus stratégiájára való globális rálátást.

A kutatás

Az AlgoRhythmics környezet egyik jellemző tulajdonsága, hogy aktív tanulást biztosít, azaz lehetővé teszi a tanulók részére az interaktív részvételt az algoritmus vizualizációs folyamatban. Ebben a kutatásban arra szerettünk volna választ kapni, hogy milyen előnyei,

illetve hátrányai lehetnek az általunk definiált nulla-, fél-, illetve teljes-interaktivitásnak online algoritmus vizualizációs környezetekben. „0-interaktivitásnak” neveztük azt a tanmenetet, amikor a tanuló egyszerűen csak megtekinti a vizualizációt. Azt a változatot, amikor a tanulóktól azt várják el, hogy csak bizonyos kulcsmomentumokban kapcsolódjanak be a vizualizációs folyamatba, „ $\frac{1}{2}$ -interaktivitásnak” („fél-interaktivitás”) tekintettük. „1-interaktivitás” (vagy „teljes interaktivitás”) a neve annak az esetnek, amikor a diákoknak a teljes animációs folyamatot le kell vezényelniük.

Ez a kutatás egyfajta folytatása az előző fejezetben bemutatott tanulmánynak. Arra a 2018/19-es tanév elején, erre pedig a 2019/20-as tanév elején került sor. A 2018/19-es tanév elsőévesein azt mértük, hogy előnyös-e, ha a diákok kétszer is megtekintik egy algoritmus két különböző vagy ugyanazon vizualizációját. Az előteszt számítógépes gondolkodást mérő feladatokon alapult, a tanulási szakasz és az utóteszt pedig a Shell rendezés köré épült. A kísérlet hasonló felépítésű volt, csak hogy a résztvevők a Shell rendezés AlgoRhythmics animációját tekintették meg kétszer és a három csoport, amelyet kialakítottunk, a második megtekintés alatt bekapcsolódott a vizualizációs folyamatba, a 0-, $\frac{1}{2}$ -, illetve 1-interaktivitási szinteknek megfelelően. A kísérletben 134 elsőéves, reálszakos hallgató vett részt, akiknek a 13.97%-volt lány. A résztvevők véletlenszerűen lettek három csoportba osztva, amelyek egyenlő szintűeknek bizonyultak mind a középiskolában programozással töltött évek átlagai alapján, mind az előteszt eredmények tekintetében.

Eredmények és kiértékelés

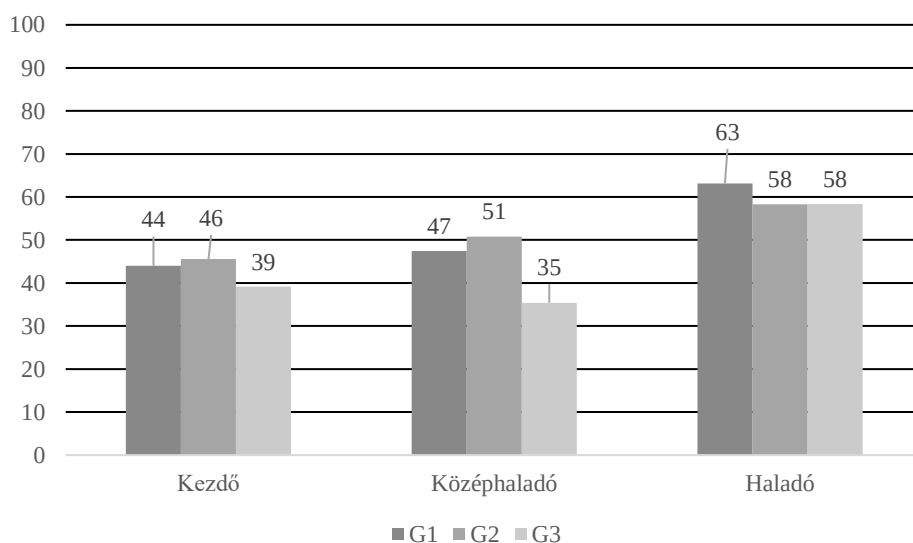
Interaktivitási szintek hatékonyságának vizsgálata

Az utóteszt kiértékelése során nem találtunk szignifikáns különbséget a három csoport között. Ily módon a kutatási kérdésünkkel kapcsolatban úgy következtethetünk, hogy nem eredményez számottevő különbséget a tanulási teljesítményben, ha nulla-, fél- vagy teljes-interaktivitást alkalmazunk. Ez összhangban van a korábbi kutatásokkal, abban az értelemben, hogy kimutatták, hogy mindhárom tanmenet típusnak megvannak az előnyei, illetve hátrányai és ezek az eredmények összecsengnek Byrne és társszerzői (Byrne, Catrambone, and Stasko 1996) nevéhez fűződő meta-analízis egyes eredményeivel.

Mivel az előtesztelés során összességében nem találtunk szignifikáns különbséget a különböző csoportok között, megvizsgáltuk a tanulók utóteszten elért teljesítményét az előzetes programozási tapasztalatok függvényében is.

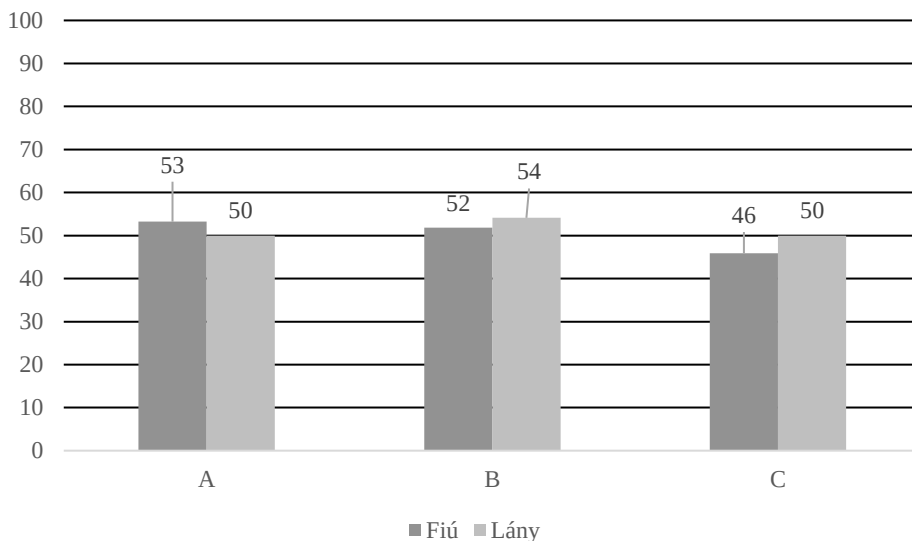
Azoknál a tanulóknál, akik nem rendelkeztek előzetes programozási ismeretekkel (0 évet) vagy csak nagyon keveset (1-2 évet) tanultak programozást, az eredmények lineáris növekedését tapasztaltuk az interaktivitás növekedésének függvényében (44%, 47%, 64% - 0 év programozás; 46%, 51%, 58% - 1 vagy 2 év programozás) (T3.1 tézis).

A haladók (3 vagy 4 év tanultak programozást) eredményeit nem befolyásolta a tanulás interaktivitási szintje (T3.2 tézis) (4. ábra).



4. ábra: A 3 programozási szinten levő diákok eredményei az utóteszt során, csoportonként

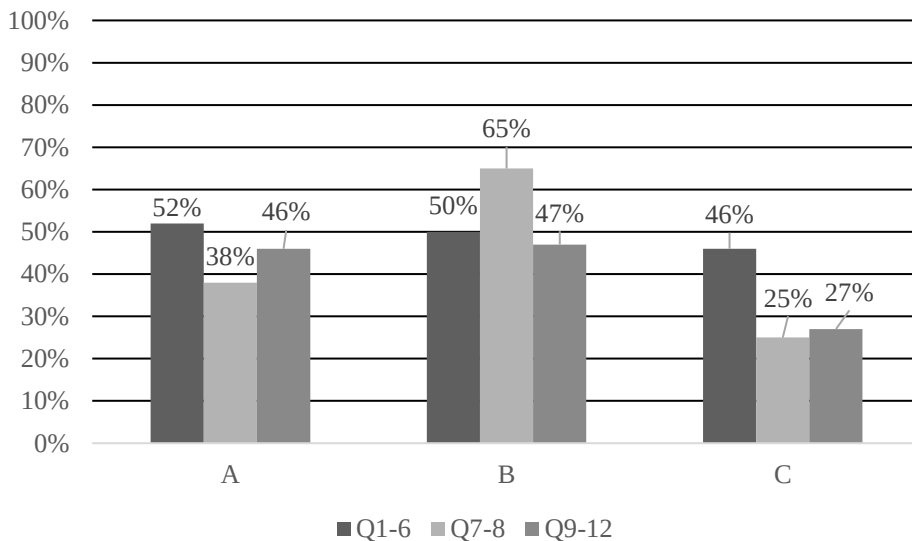
Annak ellenére, hogy az előtesztelés során a fiúk valamivel jobban teljesítettek mint a lányok, az utótesztelésen, azoknál a csoportoknál, ahol a tanítás interaktívan zajlott, a lányok eredményei a fiúk eredményei a fölé kerekedtek (5. ábra).



5. ábra: Az előteszt és utóteszt eredményei nemeként csoportokra lebontva

Az 1 vagy 2 év programozási ismerettel rendelkező tanulók válaszait a kérdések típusa szerinti felbontásban is megvizsgálva, a következő eredményekhez jutottunk:

- A 7–8. kérdések az algoritmus egy-egy adott pontján történő művelethez kapcsolódtak. Itt azok a tanulók értek el szignifikánsan jó eredményeket, akik „ $\frac{1}{2}$ interaktivitással” tanultak, vagyis előre nem ismert időpillanatokban kellett bekapcsolódjanak a vizualizációba és ez meghozta gyümölcsét. Ebben az esetben azok a tanulók teljesítettek a legrosszabbul, akik teljes, „1 interaktivitással” tanultak.
- A 9–12. kérdéseknél, amelyek szintéziskérdések voltak, azok a tanulók, akik úgymond zavartalanul el tudnak mélyülni az animációs folyamatban („0 interaktivitás”) valamint azok, akik csak bizonyos kulcsmomentumokban voltak felkérve, hogy kapcsolódjanak be a vizualizációs folyamatba a tanulás alatt („ $\frac{1}{2}$ interaktivitás”) szignifikánsan jobb eredményeket értek el a teljes, „1 interaktivitással” tanuló diákokkal szemben (T3.2 tézis) (6. ábra).



6. ábra: Az 1-6., 7-8., 9-12. kérdések válaszai az 1 vagy 2 évet tanultak esetén

Megjelent publikációk

Idegen nyelvű konferencia közlemény [5]

4. Tanári kérdéssorok algoritmus vizualizációs környezetekben

TÉZISEK

A kutatási eredmények az alábbi következtetéshez vezettek:

[T4.1] A tanár által irányított kérdés-felelet megbeszélésnek döntő szerepe lehet az algoritmusvizualizációs környezetekben rejlő potenciál teljesebb kiaknázásában.

A kutatás

Ebben a kutatásban azt vizsgáltuk, hogy mi lenne a tanár szerepe egy olyan online tanulási környezetben, mint amilyen az AlgoRythmics platform is. Arra összpontosítottunk, hogy miként járulhat hozzá a tanár a környezetben rejlő potenciál jobb kiaknázásához azáltal, hogy ügyesen kérdez.

A saját tapasztalataink és kollegáink visszajelzései is azt hangsúlyozták, hogy a diákok, amennyibe magukban használják az AlgoRythmics környezetet, csak részlegesen tudják kiaknázni a potenciált, amit magában rejt. Például azok a definíciók, amelyek próbálják körül írni a számítógépes gondolkodás fogalmát, hangsúlyozzák, hogy fontos, hogy a tanulók a tanulmányozott algoritmusok hatékonyság-vizsgálatában is elmélyüljenek (Shute, Sun &

Asbell-Clarke, 2017). Egy diák, aki úgymond autodidakta módon használja a környezetet, talán megérti a vizsgált algoritmus stratégiáját, de nem valószínű, hogy egyáltalán gondol is hatékonysági szempontokra. Viszont egy vegyes oktatási környezetben („blended learning”), a tanár rá tudja vezetni a tanulókat e szempontokra (Arthur C. Graesser, Natalie K. Person, 1994”). Persze fontos, hogy ne sérüljön az aktív tanulás elve, és a tanár megmaradjon irányítói szerepkörében azáltal, hogy támogatja a diákot, hogy egy adott irányba gondolkodjon, és olyan tartalmakat is érzékeljen, amire magától nem igazán jönne rá (Guzdial & Tew, 2006).

Egy hatékony módja annak, hogy a tanár úgy serkentse a diákot, hogy közben megmaradjon az aktív tanulás élménye az lehet, ha a tanár ügyesen kérdez. Régtől fogva elismert a kérdések értéke az oktatásban. Természetesen a kérdéseknek, hogy valóban elérjék céljukat, jól átgondoltnak kell lenniük, és az sem mindegy, hogy milyen sorrendben tesszük fel ezeket (Tofade, Elsner, & Haines, 2013).

A kutatásra a 2017-18-as tanév elején került sor. A kísérletben, ez esetben is, minden elsőéves, reálszakon tanuló hallgatót bevontunk (181 diák, 13% lány). A résztvevőket annak alapján soroltuk csoportokba, hogy hány évig tanultak programozást a középiskolai tanulmányaik alatt: A-csoport (0 évet tanultak, 27%); B-csoport (1 vagy 2 évet tanultak, 25%); C-csoport (4 évet tanultak, 48%). Az B-csoport esetében a középiskolai tanterv kitért a két alapvető keresési algoritmusra (lineáris és bináris), illetve tartalmazta néhány négyzetes időbonyolultságú rendezési algoritmus vizsgálatát, alapszinten. Ily módon, ezek a résztvevők, bár ismerhették a szóban forgó algoritmusok stratégiáját, az algoritmus-bonyolultság fogalma új volt számukra. Azon csoport tagjai, akik 4 évet tanultak (C-csoport), az algoritmusok stratégiája mellett, bonyolultsági fogalmakkal is megismerkedtek a középiskolában.

A kísérletünk első szakaszához az A-csoportot tekintettük kísérleti csoportnak, a B-t pedig kontroll csoportnak. Kíváncsiak voltunk rá, hogy megvan-e az AlgoRythmics környezetben a potenciál ahhoz, hogy az A-csoport diákjait felzárkóztassa a B-csoporthoz. A C-csoport, ebbe az elemzésben, egyfajta másodlagos kontroll csoportként lett bevonva.

A kísérlet második és harmadik szakaszában csak az A-csoport tagjai vettek részt. Arra számítottunk, hogy miután e résztvevők elmélyültek az első szakasz kérdéseiben, és aktívan részt vettek a második szakasz keretében zajló megbeszélésen, a harmadik szakaszban

szignifikánsan jobban fognak teljesíteni, mint ahogy az első szakaszban tették (e két szakasz megfelelő kérdései kapcsán).

Eredmények és kiértékelés

A háromszakaszos tanmenet során a kérdések kétféle formában is megjelentek. Az első és harmadik szakaszokban a vizualizációkhoz csatolt kérdésekben a résztvevők magukban mélyülhettek el, a második szakaszban viszont egy tanár irányította kérdés-feleletes megbeszélés révén szembesültek a diákok a tanári kérdésekkel.

Az eredmények arra mutatnak, hogy az AlgoRhythmics videók elég kifejezők ahhoz, hogy amennyiben megfelelő kérdéseket társítunk hozzájuk, a diákok kikövetkeztethessék belőlük a vizsgált algoritmus stratégiáját. Mivel a harmadik szakaszban, egy új algoritmuson, programozói előismeretekkel nem rendelkező diákok szignifikánsan jobban teljesítettek, mint az első szakaszban, ez arra enged következtetni, hogy egy a második szakaszbeli osztálytermi kérdés-feleletes megbeszélés jelentősen hozzájárulhat a diákok megértésének elmélyítéséhez.

Megjelent publikációk

Idegen nyelvű konferencia közlemény [8]

5. Számítógépes gondolkodás vizsgálata különböző korosztályú tanulóknál

TÉZISEK

[T5.1] Ha egy oktatási rendszer nem fókuszál explicite a számítógépes gondolkodás fejlesztésére, akkor a többi tantárgyak járulékos hozadékaként csak moderáltan fejlődik e képesség.

[T5.2] Megfelelő módszerekkel és eszközökkel már elemi szinten elsajátíthatók a számítógépes gondolkodás bizonyos elemei (ugyanazon hatékonysággal, mint felsőbb osztályokban), mindkét nem esetében.

A kutatás

Ebben a kutatásban azt vizsgáltuk, hogy milyen mértékben növekedik a tanulók számítógépes gondolkodása egy olyan oktatási rendszer révén, amely nem összpontosít explicite e képesség fejlesztésére. Azt is igyekeztünk felmérni, hogy milyen mértékben képesek elsajátítani, különböző korosztályú tanulók, egy számítógépes algoritmust (egy AlgoRhythmics vizualizáció segítségével).

Ahelyett, hogy egy általános számítógépes gondolkodás tesztet alkalmazzunk, mint például amelyet Román és munkatársai (2018) dolgoztak ki, inkább úgy döntöttünk, hogy úgy mérjük le a diákok számítógépes gondolkodását, hogy egy olyan feladattal szembesítjük őket, melynek megoldása feltételezi, valamilyen szinten, e képesség birtoklását. Mivel a számítógépes gondolkodás fogalma az algoritmikus gondolkodás fogalmából nőtt ki, egy olyan tanulási környezetet hozunk létre, amely háttéréül egy számítógépes algoritmus (lineáris keresés) szolgált. A feladat megszövegezésekor odafigyeltünk arra, hogy a 3. osztályos tanulók számára is érthető és érdekes legyen. Az alapelveket, amelyet figyelembe vettünk, Comenius a következőképpen fogalmazta meg: „A gyermekekkel megtaníthatók a felnőttkor tevékenységei, csak gyermeki módon” (Comenius 1992: Pampedia, IV.rész, ford. Bollók János).

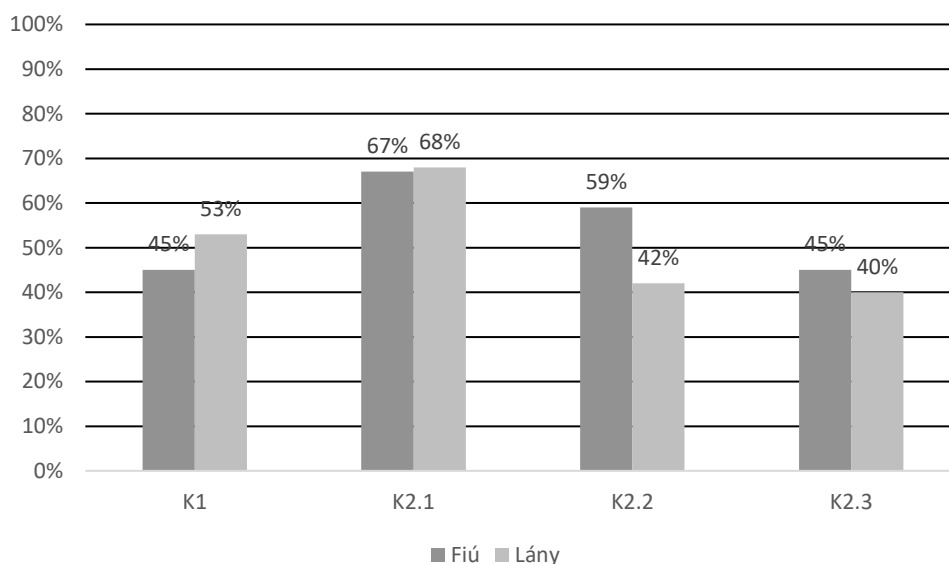
E kutatásra a 2018-2019-es tanév első félévében került sor a Bolyai Farkas Elméleti Líceumban, illetve a marosvásárhelyi Művészeti Iskolában. A kísérletet ugyancsak háromszakaszosra terveztük. Az első szakaszban igyekeztük felmérni azt, hogy milyen szintű lehet, az aktuális tantervük eredményeként, a résztvevők számítógépes gondolkodása, azaz milyen mértékben érznek rá a lineáris keresési stratégiára. A második szakaszban (tanulási fázis) a tanulók megtekintették ezen algoritmus tánckoreográfia ábrázolását. A harmadik szakaszban a résztvevők olyan kérdésekkel szembesültek, amelyek feltárhatták, hogy mennyire értették meg a vizualizált algoritmust.

A kísérlet alanyai 3., 5., 7. és 9. évfolyamos tanulók voltak, összesen 214 résztvevő (56%-a lány). Mivel Romániában elemi szinten csak elméleti és művészeti szakirány létezik, ezért esett a választás egy (e)lméleti (E-iskola) és egy (m)űvészeti (M-iskola) iskolára. Nyolc osztályt vontunk be a kísérletbe, mindenik évfolyamról, mindkét iskolából, egyet-egyet. Ezekre az osztályokra, a következőkben úgy utalunk, mint 3E, 3M, 5E, 5M, 7E, 7M, 9E és 9M osztályok.

Eredmények és kiértékelés

Az eredmények azt mutatták, hogy a jelenlegi tanterv tantárgyainak járulékos hozadékaként létezik egyfajta növekedés, de ez viszonylag lassú. Kétévenkénti eltolásban nem mutatható ki, viszont négyévenkéntiben igen. Egy érdekes észrevételünk, hogy úgy tűnik, hogy a művészet iskola tanterve inkább hozzájárul a diákok számítógépes gondolkodásának fejlődéséhez, mint az elméleti iskoláé.

Egy további következtetése a jelen fejezetnek az, hogy megfelelő módszerekkel és eszközökkel a különböző korosztályú diákok egyformán jól megtaníthatók olyan alapvető, számítógépes gondolkodáshoz kapcsolódó fogalmakra, mint amilyen a lineáris keresési algoritmus. Az eredmények azt is igazolták, hogy e tekintetben nincs különbség fiúk és lányok között (7. ábra).



7. ábra: A kérdésekre adott válaszok feldolgozása nemenként

Megjelent publikációk

Magyar nyelvű konferencia közlemény [1]

Összefoglaló

Az AlgoRythmics fejlesztés 2007-ben indult, viszont 2016-ban új lendületet kapott. Ekkortól vagyok jómagam is része a kutatócsoportnak, és a dolgozatban bemutatott kutatások is azóta zajlottak. Az első tánckoreográfiákat 2011-ben tette közzé a kutatócsoport. Ezek hat számítógépes algoritmust illusztráltak: kiválasztó, beszűrő, buborék, Shell, összefésülő és gyors rendezés. A nemzetközi visszhang nagyon kedvező volt, és rengeteg kérés érkezett a kupac-rendezés vizualizációjának elkészítésére, illetve arra, hogy terjesszük ki a módszert más algoritmus csoportokra is. Az én feladatomban volt, hogy főszerepet vállaljak ebben. Sikertült számos új elemmel gazdagítani az azóta született négy új algoritmus ábrázolást (kupac-rendezés, lineáris és bináris keresés, visszalépéses keresés a négy-királynő

problémára). Például, az előző videóknál mindig egy-egy táncos képviselte a számsorozat elemeit, a kupac-rendezés illusztrációban viszont táncos párok töltik be a szerepet. Egy további újdonság ebben a videóban, hogy a számsorozat egydimenziós ábrázolása mellett megjelenik a kétdimenziós ábrázolás is, hiszen a kupac úgy mond kinyílik bináris fává. Új tánc stílusokat vontunk be, ugyanis a lineáris és bináris keresések flamenco tánccal, a visszalépéses keresési algoritmus pedig balett koreográfiával van illusztrálva. Az előző kutatások arra mutattak, hogy növelheti a vizualizációk hatékonyságát, ha a számsorozatok „rejtetten vannak ábrázolva”. Ennek következményeként a lineáris és bináris keresések esetében a számsorozatot képviselő lányok csak a hátukon viselik a számokat.

A kutatócsoport már a kezdetektől tudatában volt, hogy a tánc-koreográfia illusztrációk nem helyettesítik az algoritmusok absztrakt animációkkal való ábrázolását. Az is világos volt, hogy ahhoz, hogy a vizualizációkban rejlő potenciál kellőképpen kiaknázható legyen, szükséges egy megfelelő tanulási környezetet építeni köréjük. Már 2012-ben készült egy webes alkalmazás e célból, amely a videók mellé számítógépes animációkat társított, és lehetővé tette az ezekkel való interaktív tanulást. Ahogy készültek el az újabb algoritmus illusztrációk, úgy vált nyilvánvalóvá, hogy az AlgoRhythmics webes környezet is megújításra szorul. Néhány fejlesztés, amelyben kulcsszerepet vállaltam a következők: legyen beállítható, hogy milyen interaktivitási szinttel szeretné a környezetet a felhasználó használni; nyújtson segítséget a környezet az algoritmus megértése, és a számítógépes kód elkészítése közt tátongó szakadék áthidalásához; tegye lehetővé a környezet a felhasználók teljesítményének mérését.

IRODALOMJEGYZÉK ÉS HIVATKOZÁSOK – BIBLIOGRAPHY AND REFERENCES

Ahadi, A., Lister, R., Lal, S., Leinonen, J., & Hellas, A. (2017, January). Performance and Consistency in Learning to Program. In *Proceedings of the Nineteenth Australasian Computing Education Conference* (pp. 11-16). ACM.

Aho, A. V. (2011). Computation and computational thinking. *Ubiquity Symposium*. DOI: 10.1145/1895419.1922682

Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, 55(7), 832–835.

Ainsworth, S., & VanLabeke, N. (2004). Multiple forms of dynamic representation. *Learning and instruction*, 14(3), 241-255.

AlgoRhythmicenvironment. (2020). <https://algorhythmics.ms.sapientia.ro/>.

Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661-670.

Ayres, P., & Paas, F. (2007). Making instructional animations more effective: A cognitive load approach. *Applied Cognitive Psychology*, 21, 695–700.

Berney, S., & Betrancourt, M. (2016). Does animation enhance learning? A meta-analysis. *Computers & Education*, 101, 150–167.

Butcher, K. R. (2006). Learning from text with diagrams: Promoting mental model development and inference generation. *Journal of Educational Psychology*, 98(1), 182.

Byrne, M. D., Catrambone, R., & Stasko, J. T. (1996). Do algorithm animations aid learning? Georgia Institute of Technology.

Byrne, P., & Lyons, G. (2001, June). The effect of student attributes on success in programming. In *ACM SIGCSE Bulletin* (Vol. 33, No. 3, pp. 49-52). ACM.

Carney, R. N., & Levin, J. R. (2002). Pictorial illustrations still improve students' learning from text. *Educational psychology review*, 14(1), 5-26.

Castro-Alonso, J. C., Ayres, P., & Paas, F. (2014). Dynamic visualisations and motor skills. In *Handbook of human centric visualization* (pp. 551-580). Springer, New York, NY.

Castro-Alonso, J. C., Ayres, P., Wong, M., & Paas, F. (2018). Learning symbols from permanent and transient visual presentations: Don't overlay the hand. *Computers & Education*, 116, 1–13.

Comenius (1992): *Pampedia*. (IV.rész, ford. Bollók János), B-A-Z Levéltár, Sárospatak. IX. fejezet, A kisgyermekkor iskolája, az anyaöl, vagyis az emberi sarjnak a születéstől körülbelül hatéves koráig tartó gondos nevelése. 68-80.

de Koning, B. B., & Tabbers, H. K. (2011). Facilitating understanding of movements in dynamic visualizations: An embodied perspective. *Educational Psychology Review*, 23(4), 501–521.

Denning, P. J. (2009). The profession of IT Beyond computational thinking. *Communications of the ACM*, 52(6), 28-30.

Denning, P. J. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, 60(6), 33-39.

Evans, G. E., & Simkin, M. G. (1989). What best predicts computer proficiency? *Communications of the ACM*, 32(11), 1322-1327.

Feaster, Y., Ali, F., Zhai, J., & Hallstrom, J. O. (2014, June). Serious toys: three years of teaching computer science concepts in K-12 classrooms. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 69-74). ACM.

Fritz, C. O., Morris, P. E., Bjork, R. A., Gelman, R., & Wickens, T. D. (2000). When further learning fails: Stability and change following repeated presentation of text. *British Journal of Psychology*, 91(4), 493-511.

Goldstone, R. L., & Son, J. Y. (2005). The transfer of scientific principles using concrete and idealized simulations. *The Journal of the Learning Sciences*, 14(1), 69-110.

Graesser, A. C., & Person, N. K. (1994). Question asking during tutoring. *American educational research journal*, 31(1), 104-137.

- Guzdial, M., & Tew, A. E. (2006, September). Imagineering inauthentic legitimate peripheral participation: an instructional design approach for motivating computing education. In *Proceedings of the second international workshop on Computing education research* (pp. 51-58). ACM.
- Höffler, T. N., & Leutner, D. (2007). Instructional animation versus static pictures: A meta-analysis. *Learning and Instruction*, 17(6), 722-738.
- Höffler, T. N., Leutner, D. (2007). Instructional animation versus static pictures: A meta-analysis. *Learning and Instruction*, 17(6), 722-738.
- Jarc, D. J., Feldman, M. B., & Heller, R. S. (2000). Assessing the benefits of interactive prediction using web-based algorithm animation courseware. *ACM SIGCSE Bulletin*, 32(1), 377-381.
- Katai, Z. (2014, June). Selective hiding for improved algorithmic visualization. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 33-38). ACM.
- Káta, Z. (2015). The challenge of promoting algorithmic thinking of both sciences-and humanities-oriented learners. *Journal of Computer Assisted Learning*, 31(4), 287-299.
- Katai, Z., Toth, L., & Adorjani, A. K. (2014). Multi-Sensory Informatics Education. *Informatics in Education*, 13(2), 225-240.
- Káta, Z., Juhász, K., & Adorjáni, A. K. (2008). On the role of senses in education. *Computers & Education*, 51(4), 1707-1717.
- Korhonen, A., & Malmi, L. (2000). Algorithm simulation with automatic assessment. *ACM SIGCSE Bulletin*, 32(3), 160-163.
- Lau, W. W., & Yuen, A. H. (2009). Exploring the effects of gender and learning styles on computer programming performance: implications for programming pedagogy. *British Journal of Educational Technology*, 40(4), 696-712.
- Lewandowski, G., Bouvier, D. J., Chen, T. Y., McCartney, R., Sanders, K., Simon, B., & VanDeGrift, T. (2010). Commonsense understanding of concurrency: computing students and concert tickets. *Communications of the ACM*, 53(7), 60-70.
- Lin, L., & Li, M. (2018). Optimizing learning from animation: Examining the impact of biofeedback. *Learning and Instruction*, 55, 32-40.
- Lowe, R. K. (1999). Extracting information from an animation during complex visual learning. *European Journal of Psychology of Education*, 14, 225-244.
- Lowe, R. K., & Boucheix, J.-M. (2017). A composition approach to design of educational animations. In R. Lowe, & R. Ploetzner (Eds.). *Learning from dynamic Visualizations: Innovations in research and application* (pp. 5-30). Berlin: Springer.
- Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014, June). Computational thinking in K-9 education. In *Proceedings of the working group reports of the 2014 on innovation & technology in computer science education conference* (pp. 1-29). ACM.
- Mayer, R. E. (2005). Cognitive theory of multimedia learning. In R. E. Mayer (Ed), *The Cambridge handbook of multimedia learning* (pp. 31-48). Cambridge University Press.
- Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist*, 38(1), 43-52.
- McCartney, R., Bouvier, D. J., Chen, T. Y., Lewandowski, G., Sanders, K., Simon, B., & VanDeGrift, T. (2009, August). Commonsense computing (episode 5): algorithm efficiency and balloon testing. In *Proceedings of the fifth international workshop on Computing education research workshop* (pp. 51-62). ACM.
- Moreno, R., Ozogul, G., & Reisslein, M. (2011). Teaching with concrete and abstract visual representations: Effects on students' problem solving, problem representations, and learning perceptions. *Journal of Educational Psychology*, 103, 32-47.
- Naps, T. L., Eagan, J. R., & Norton, L. L. (2000, May). JHAVÉ—an environment to actively engage students in Web-based algorithm visualizations. In *ACM SIGCSE Bulletin* (Vol. 32, No. 1, pp. 109-113). ACM.
- Nugteren, M. L., Tabbers, H. K., Scheiter, K., & Paas, F. (2014). Simultaneous and Sequential Presentation of Realistic and Schematic Instructional Dynamic Visualizations. In *Handbook of human centric visualization* (pp. 605-622). Springer, New York, NY.

- Paas, F., & Sweller, J. (2012). An evolutionary upgrade of cognitive load theory: Using the human motor system and collaboration to support the learning of complex cognitive tasks. *Educational Psychology Review*, 24(1), 27-45.
- Paas, F., & Sweller, J. (2012). An evolutionary upgrade of cognitive load theory: Using the human motor system and collaboration to support the learning of complex cognitive tasks. *Educational Psychology Review*, 24(1), 27-45.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc..
- Pukánszky Béla, N. (n.d.). *Neveléstörténet*. Bp., Nemzeti Tankvk., 1997. 185. o
- Rau, M. A. (2013). *Conceptual learning with multiple graphical representations: Intelligent tutoring systems support for sense-making and fluency-building processes* (Unpublished doctoral dissertation). Carnegie Mellon University, Pittsburgh, PA.
- Rau, M. A. (2017). Conditions for the effectiveness of multiple visual representations in enhancing STEM learning. *Educational Psychology Review*, 29(4), 717-761.
- Rau, M. A., & Matthews, P. G. (2017). How to make 'more' better? Principles for effective use of multiple representations to enhance students' learning about fractions. *ZDM*, 49(4), 531-544.
- Rau, M. A., Aleven, V., & Rummel, N. (2013). Interleaved practice in multi-dimensional learning tasks: Which dimension should we interleave?. *Learning and Instruction*, 23, 98-114.
- Rau, M. A., Michaelis, J. E., & Fay, N. (2015). Connection making between multiple graphical representations: A multi-methods approach for domain-specific grounding of an intelligent tutoring system for chemistry. *Computers & Education*, 82, 460-485.
- Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior*, 72, 678-691.
- Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2018). Can computational talent be detected? Predictive validity of the Computational Thinking Test. *International Journal of Child-Computer Interaction*, 18, 47-58.
- Roy, M., & Chi, M. T. (2005). The self-explanation principle in multimedia learning. *The Cambridge handbook of multimedia learning*, 271-286.
- Scheiter, K., Gerjets, P., Huk, T., Imhof, B., & Kammerer, Y. (2009). The effects of realism in learning with dynamic visualizations. *Learning and Instruction*, 19(6), 481-494.
- Settle, A., Goldberg, D. S., & Barr, V. (2013, July). Beyond computer science: computational thinking across disciplines. In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (pp. 311-312). ACM.
- Shaffer, C. A., Cooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S., & Edwards, S. H. (2010). Algorithm visualization: The state of the field. *ACM Transactions on Computing Education (TOCE)*, 10(3), 9.
- Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22, 142-158.
- Simon, B., Chen, T. Y., Lewandowski, G., McCartney, R., & Sanders, K. (2006, September). Commonsense computing: what students know before we teach (episode 1: sorting). In *Proceedings of the second international workshop on Computing education research* (pp. 29-40). ACM.
- Simon, B., Kinnunen, P., Porter, L., & Zazkis, D. (2010, June). Experience report: CS1 for majors with media computation. In *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education* (pp. 214-218). ACM.
- Stout, J., & Tamer, B. (2016, February). Collaborative learning eliminates the negative impact of gender stereotypes on women's self-concept. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education* (pp. 496-496). ACM.
- Sweller, J., Van Merriënboer, J. J., & Paas, F. G. (1998). Cognitive architecture and instructional design. *Educational psychology review*, 10(3), 251-296.
- Sweller, J., Clark, R., & Kirschner, P. (2010). Teaching general problem-solving skills is not a substitute for, or a viable addition to, teaching mathematics. *Notices of the American Mathematical Society*, 57(10), 1303-1304.

Tew, A. E., McCracken, W. M., & Guzdial, M. (2005, October). Impact of alternative introductory courses on programming concept understanding. In *Proceedings of the first international workshop on Computing education research* (pp. 25-35). ACM.

Tofade, T., Elsner, J., & Haines, S. T. (2013). Best practice strategies for effective use of questions as a teaching tool. *American journal of pharmaceutical education*, 77(7).

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433.

Tversky, B., Morrison, J. B., & Betrancourt, M. (2002). Animation: can it facilitate?.

Werth, L. H. (1986). Predicting student performance in a beginning computer science class (Vol. 18, No. 1, pp. 138-143). ACM.

Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725.



Nyilvántartási szám: DEENK/26/2021.PL
Tárgy: PhD Publikációs Lista

Jelölt: Osztián Erika

Doktori Iskola: Matematika- és Számítástudományok Doktori Iskola

A PhD értekezés alapjául szolgáló közlemények

Idegen nyelvű tudományos közlemények külföldi folyóiratban (1)

1. Kátai, Z., Osztián, E.: Improving AlgoRhythmic Teaching-Learning Environment by Asking Questions.
Int. J. Instr. 14 (2), 27-44, 2021. ISSN: 1694-809X.

Magyar nyelvű konferencia közlemények (4)

2. Osztián, E., Kátai, Z.: Az algoritmikus gondolkodás vizsgálata különböző korosztályú tanulóknál, e-learning tesztelési környezetben.
In: INFODIDACT 2019. Szerk.: Szilágyi Péter, Zsakó László, Webdidaktika az Oktatásért és az Információs Társadalomért Alapítvány, Budapest, 1-14, 2020. ISBN: 9786158080837
3. Kátai, Z., Osztián, E.: Algoritmikus gondolkodás fejlesztése különböző korosztályú és szakos tanulóknál.
In: ENELKO 2018 XIX. Nemzetközi Energetika-Elektrotechnika Konferencia, SzámOkt 2018 XXVIII. Nemzetközi Számítástechnika és Oktatás Konferencia. Szerk.: Biró Károly-Ágoston, Sebestyén-Pál György, Szabó Loránd, Erdélyi Magyar Műszaki Tudományos Társaság, Tusnádfürdő, 225-230, 2018, (ISSN 1842-4546)
4. Kátai, Z., Osztián, E., Vekov, G. K., Osztián, P. R.: Algo-Ritmika: új dimenziók.
In: INFODIDACT 2017. Szerk.: Szilágyi Péter, Zsakó László, Webdidaktika az Oktatásért és az Információs Társadalomért Alapítvány, Budapest, 1-11, 2017. ISBN: 9786158080813
5. Osztián, P. R., Kátai, Z., Osztián, E., Vekov, G. K.: Algoritmika és művészet, avagy kupacrendezés kalotaszegi módra = Algorithms and art Heapsort with traditional Transylvanian dance.
In: ENELKO 2017 XVIII. Nemzetközi Energetika-Elektrotechnika Konferencia, SzámOkt 2017 XXVII. Nemzetközi Számítástechnika és Oktatás Konferencia, Kolozsvár, 2017. október 12-15. Szerk.: Biró Károly-Ágoston, Sebestyén-Pál György, Erdélyi Magyar Műszaki Tudományos Társaság, Cluj-Napoca, 199-203, 2017, (ISSN 1842-4546)





Idegen nyelvű konferencia közlemények (4)

6. Káta, Z., Osztián, E., Lőrincz, B.: The effect of multiple graphical representations on learning algorithms.

Turk. Online J. Educ. Technol. 2018 (2), 703-708, 2018. ISSN: 2146-7242.

7. Osztián, E., Káta, Z., Vekov, G. K.: Multi-dimensional expansion of Algo-Rhythms.

Turk. Online J. Educ. Technol. 2017 (3), 573-578, 2017. ISSN: 2146-7242.

8. Káta, Z., Osztián, E., Vekov, G. K.: Promoting computational thinking by artistically enhanced algorithm visualization.

In: INFODIDACT 2016. Szerk.: Szilágyi Péter, Zsakó László, Webdidaktika az Oktatásért és az Információs Társadalomért Alapítvány, Budapest, 1-12, 2016. ISBN: 9786158060806

9. Osztián, E.: Stimularea creativității tehnico - științifice la disciplinele Tehnologia informației și a comunicațiilor și Informatică prin programarea roboților Lego.

Simpoz. Naț. Edmond Nicolau, 9, 222-232, 2015. EISSN: 2069-9530.

Magyar nyelvű absztrakt kiadványok (1)

10. Káta, Z., Osztián, E., Vekov, G. K.: Számítógépes gondolkodás fejlesztése vegyes oktatási környezetben.

In: XXIII. Multimedia in Education Conferences, Presa Universitară Clujeană, Cluj-Napoca, 104, 2017. ISBN: 9786063701832





További közlemények

Időgen nyelvű konferencia közlemények (2)

11. Osztián, P. R., Kátai, Z., **Osztián, E.**: Algorithm Visualization Environments: degree of interactivity as an influence on student-learning.
In: 2020 IEEE Frontiers in Education Conference (FIE), IEEE, dPiscataway, 1-8, 2020, (ISSN 2377-634X) ISBN: 9781728189611
12. Nagy, E. J., Osztián, P. R., Cosma, C., Kátai, Z., **Osztián, E.**: Looking for the Optimal Interactivity Level in the AlgoRythmics Learning Environment.
In: EdMedia + Innovate Learning : 2019 conference, Ed.: Theo Bastiaens, Association for the Advancement of Computing in Education, Waynesville, 106-114, 2019. ISBN: 9781939797421

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

Debrecen, 2021.01.21.



* * *

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

**Interactive, multi-sensory electronic learning
environment for the development of computer
thinking**

by Osztián Erika

Supervisor: Dr. Fazekas Gábor



UNIVERSITY OF DEBRECEN
Doctoral School of Mathematical and Computational Sciences

Debrecen, 2021

INTRODUCTION

Nowadays it is commonly accepted that computational thinking (CT) is an essential mindset for all students of the digital era. Although the phrase computational thinking was introduced by Seymour Papert in 1980 (Papert, 1980), it was brought to the forefront of the computer science (CS) education community only in 2006 by Jeannette Wing (Wing, 2006). Wing describes CT as a formative skill on a par with reading, writing and arithmetic, and emphasizes that everyone, not just computer scientists, would be eager to learn and use. Since the concept of CT can be defined as the thought process involved in formulating problems so that “their solutions can be represented as computational steps and algorithms” (Aho, 2012), a natural way to address the “CT for all” issue is computing education for all. For example, Echeverría, Cobos, Machuca and Claros (2017) emphasize that computing is a skill required for any engineering field.

A possible approach to make computing education attractive for different category of learners (including K-12 learners and non-CS majors) is contextualization (Guzdial, 2010). For example, in the case of non-CS majors the context should be related to the major field of the students. Since developing differentiated teaching–learning strategies may involve substantial additional costs, some scholars have tried to find a context that is appealing to most students. A promising candidate for this “common denominator role” could be arts (Tew, McCracken, & Guzdial, 2005; Guzdial & Tew, 2006; Simon et al., 2010; Daily et al., 2014; Wood, Muhl, & Hicks, 2016).

The AlgoRythmics (2019) learning environment was designed along this approach. Since music and dance are relatively close to most young people, this environment visualizes ten basic computer algorithms (searching and sorting) by professional dance choreographies (folkdance, flamenco, ballet). The videos are also accessible by the AlgoRythmics (2011) YouTube channel and they have got millions of views. Beside the motivating arts-science combination, another advantage of the AlgoRythmics videos is that they activate multi-sensory learning. In addition, the algorithms are illustrated by human movement. Recent research results on the so-called human movement effect emphasize that observing human movements (or producing our own body movement) can be cognitively beneficial (Castro-Alonso, Ayres, Wong, & Paas, 2018).

The AlgoRythmics research group underline that they were aware from the outset that, although videos can have their own strengths, they do not replace abstract animations, they

can only complement them (Káta, 2015). According to Katai, Toth and Adorjani (2014) it was from this consideration that the AlgoRythmics environment (2011) was created in which an interactive computer animation is associated to each video.

In conclusion it can be stated that AlgoRythmics platform is a multisensory, interactive e-learning environment to promote the CT of different category of learners. The AlgoRythmics project started in 2007, the first six video was created in 2011 and the first version of the attached interactive learning environment was developed in 2012. I joined the research group in 2016. Since then I had a leading role in the designing and implementing process of the four new dance choreographies (linear and binary search, heap sort, and backtracking algorithm for the four-queen problem) and I have been actively participating in the developing process of the new version of the AlgoRythmics interactive web-application. All the five studies that I am going to describe in the followings were implemented in this renewed AlgoRythmics environment.

NEW RESULTS

1. Study 1: Schematic and human movement effect enhanced realistic algorithm visualization

Teaching learning algorithms is an important issue in Computer Science education and algorithms are central to computational thinking too (Denning, 2009). Turing (1950) emphasizes that understanding thoroughly how computer algorithms work assumes that learners are able to build “a clear mental picture of the state of the machine at each moment in the computation”. Graphical visualizations that illustrate algorithms’ procedural behavior are meant to support this internal building process (Káta, 2014). Since algorithms are dynamic processes, dynamic visualizations such as animation and video are increasingly used in learning environments for studying computer algorithms (de Koning & Tabbers, 2011; Shaffer et al., 2010). In this study we investigated the relative effectiveness of schematic animations and realistic videos in the context of algorithm visualization.

Schematic or/and realistic algorithm visualization

Instructional dynamic visualizations can be classified as schematic or realistic representations (Nugteren et al., 2014). During the last decades some relevant studies (from different fields of education) have focused on the relative value of these type of visualizations (Denning, 2017; Nugteren et al., 2014; Scheiter, 2009). However, the topic of algorithm visualization has been poorly researched from this perspective. A possible reason

could be that computer algorithms are inherently abstract entities, which lack any tangible real-world representation. Consequently, schematic animations are commonly used for illustrating computer algorithms. On the other hand, there are representations, especially in the context of unplugged CS education, that are closer to realistic visualizations. A relevant example in this sense is the AlgoRhythmic (2011) environment which includes realistic dance choreography illustrations.

Prior studies that compared the effectiveness of these two types of visualizations conclude that both types could have advantages depending on the learning outcomes that need to be accomplished (Nugteren et al., 2014). For example, based on the coherence principle of multimedia learning, a schematic animation may support learners in focusing on the important aspects of the visualization because it contains less irrelevant/distracting elements (Butcher, 2006; Mayer & Moreno, 2003). On the other hand, according to Goldstone and Son (2005), two benefits of realistic visualizations that might be extended to the AlgoRhythmic videos too are: (1) they can more easily be remembered (being more concrete); (2) they promote intrinsic motivation (could be more appealing).

Human movement effect

Surprisingly, the studies that researched whether dynamic visualizations aid learners' understanding of dynamic phenomena have reported mixed results (Ainsworth & VanLabeke, 2004). The most commonly raised reason why animations are not found to be consistently effective is their transient nature (Atmatzidou & Demetriadis, 2016; Grover & Pea, 2013). However, interestingly, dynamic visualizations have been found to be consistently superior to static visualizations when learning with animation involved human movement (Castro-Alonso, Ayres & Paas, 2014). In this special case, it seems that the mirror neuron system assists working memory in coping with transitory information.

Accordingly, a particular strength of the AlgoRhythmic videos is that they illustrate the basic operations of the algorithms by human movements. Recent research results emphasize that observing human movements can be cognitively beneficial (Castro-Alonso, Ayres & Paas, 2014). According to Van Gog et al. (2009), this so-called "human movement effect" (HME) (Paas & Sweller, 2012) enhances dynamic visualizations by counteracting the drawback resulted from their transient nature.

The majority of the prior studies in the field of realistic versus schematic representations report supporting evidence in the favor of schematic visualization. For example, Scheiter et

al. (2009) analyzed learning settings that combined realistic and schematic dynamic visualizations in the context of biology education. Results showed that participants from the realistic+realistic (the same visualization presented in succession) condition performed significantly worse than their colleagues from the other three conditions (schematic+schematic, schematic+realistic, realistic+schematic). More recently, the authors of the (Nugteren et al., 2014) study report similar results. They investigated the following learning settings: schematic-only, realistic-only, sequential schematic+realistic, simultaneous schematic+realistic. Again, the realistic-only condition scored significantly lower than the other three conditions. These findings are in accordance with the statement Tversky, Morrison and Betrancourt (2001) made that animations should contain minimal realism, because even appealing realistic details may obstruct comprehension of the relevant movements of dynamic visualizations.

But what if the realistic animation includes visualization of human movements (HME-realistic)? Can they benefit from the HME to the point that they overcome (or be at least equally as effective as) their schematic counterparts? This study focuses on this topic in the context of the AlgoRhythmics learning environment.

Schematic computer animations and realistic dance choreography videos, side by side, in the AlgoRhythmics environment

AlgoRhythmics animations belong to the category of schematic dynamic visualization, characterized by their coherent style. Any array to be searched or sorted is represented by a series of rectangles. The two key operations (compare, swap) are illustrated consistently in the same way and the elements that have reached their final position (in the case of sorting algorithms) change their color. On the other hand, each dance choreography has unique characteristics rooted in the diversity of the dance styles. A common feature is that the number sequences are represented by dancers. In the case of linear and binary search algorithms the cells of the arrays that store the numbers are represented by chairs with the dancers sitting on them. In other videos these arrays are illustrated by graphical elements added via a video editing software. The lack of a coherent style is also due to the compare and swap operations each being illustrated by their own specific dance style.

Höffler and Leutner (2007) in their meta-analysis in the field of “instructional animation versus static pictures” make a distinction between representational animations (the topic to be learned is explicitly depicted in the animation) and decorational animations (the primary

instructional function of the animation is to motivate the learner) (Carney & Levin, 2002). They found that animations are specifically superior to static pictures when the visualization plays a representational role (the depicted motion in the animation explicitly refers to the topic to be learned).

Based on this terminology we identified two types of realistic/artistic elements in the AlgoRhythmics videos: (1) decorative elements (specific costumes, music, male/female dancers, etc.) and (2) representational elements (specific dance steps illustrating the key operations of the algorithms). In accordance with this, we classified the potentially distracting elements in two categories too: (1) decorative distractors and (2) representational distractors. All decorative elements could be decorative distractors. We are talking about representational distractors when realistic/artistic elements shadow the link (the one to one relationship) between the key operations of the algorithm and the corresponding dance steps. We were particularly interested in the influence these distracting elements may have on the effectiveness of the AlgoRhythmics videos.

Theses

In extension to the above literature review we report the following research results:

[T1.1] Students assigned to the “realistic condition” will not perform less well than their counterparts from the “schematic condition group” (for example, because of the HME);

[T1.2] For the members of the “realistic condition group” it will be easier to distinguish between the two algorithms (since as realistic visualizations, they can more easily be remembered);

[T1.3] It is not so much the decorative elements, but rather the representational distractors that can obstruct the comprehension of the relevant movements of the realistic algorithm visualizations.

Details about the implemented experiment and the performed statistical analysis

Two algorithms were selected for this analysis: Bubble sort and Selection sort. Both algorithms can be perceived as a succession of comparing and comparing+swapping operations. During the Selection sort video the comparison operations are illustrated consistently by the same dance steps (regardless whether, or not, they are combined with a swapping operation). On the other hand, in the case of the Bubble sort video the combined comparing+swapping scenes do not include clearly separable comparing and swapping

phases. Because of this particularity we considered that the Bubble sort choreography contains not only decorative but also representational distractors.

We chose the Pre-test/Post-test control group experimental design. The experimental group (group-V) was presented with the realistic visualizations of the algorithms (dance choreography (v)ideos) and the control group (group-A) with the schematic ones (abstract computer (a)nimations). The study was conducted at the Sapientia Hungarian University of Transylvania at the beginning of 2019-20 academic year. All the 160 first year students (aged from 19 to 22 years) enrolled in science oriented programmes were invited to participate in the experiment. Subjects were randomly assigned to the two groups.

We organized the pre-test during the first week of the school year. The pre-test assignments were based on two, apparently CS free, tasks. The experiment took place during the second week of the semester. After a brief introduction students were engaged in the following learning-testing session (two learning and one testing phase):

- (phase-1) members of group-V/A were invited to watch the dance-choreography/computer-animation of the bubble sort algorithm, and;
- (phase-2) members of group-V/A were invited to watch the dance-choreography/computer-animation of the selection sort algorithm, and;
- (phase-3) both groups were invited to answer 13 questions (using the Socrative classroom app) regarding each algorithm.

Statistical analysis was performed using R statistical software. We used ANOVA to check whether the two groups had comparable prior knowledge at the beginning of the experiment. Results showed that the two conditions did not differ significantly on prior knowledge (group-V: 68%, group-A: 69%; $F(1,114)=0.1$, $p=0.74>0.05$). With respect to both algorithms the post-test scores analyzed with ANCOVA. The independent variable was the instructional condition (realistic vs. schematic visualization), and the dependent variable was the corresponding post-test result. Prior knowledge (pre-test scores) was used as a covariate. We found that while there were no significant differences in the Bubble sort scores (group-V: 65%, group-A: 68%; $F(1,113)=0.78$, $p=0.37>0.05$), in the case of Selective sort the experimental group performed significantly better than the control one (group-V: 58%, group-A: 45%; $F(1,113)=9.52$, $p=0.002<0.05$; partial $\eta^2 = 0.07$).

These results support our first hypothesis that group-V will not perform less well than group-A. Although this result is contrary to previous studies, such as (Nugteren et al., 2014; Scheiter et al., 2009), it is in line with recent results in HME (as detailed above). According to Goldstone and Son (2005), a possible advantage of realistic visualizations is that they can be remembered easier. In the case of AlgoRhythmics videos it is not an exaggeration to state that, because of the specific artistic elements, each dance choreography has a kind of personality. To answer phase-3 questions (which alternated between the two algorithms) participants had to recall and clearly separate the two sorting strategies in their mind. This would be easier for group-V, which could explain why this group performed better than the other on the second algorithm. Since the animations contained the same “decorational elements”, group-A students could more easily confuse the two algorithms. This result, according to our second hypothesis, emphasizes that one of the strengths of the AlgoRhythmics videos might be that they can more easily be remembered.

As a next step, we carefully analyzed students’ responses with respect to each question or group of questions. With respect to our third hypothesis the results reveal that while the HME may have the potential to compensate the distraction effect generated by the artistically enhanced decorational elements, representational distractors may affect (adversely) the effectiveness of realistic visualizations. This is plausible since these distracting elements may directly affect the movements that illustrate the key operations of the algorithm.

In order to investigate whether males and females from the two groups (experimental vs. control) performed equally well, post-test two way ANOVAs were conducted. The two independent variables were the instructional condition (realistic vs. schematic visualization) and the gender, while the dependent variable was students’ post scores. No interactions were detected in the case of both algorithm (Bubble sort: $p=0.91>0.05$; Selection sort: $p=0.65>0.05$). These results are consistent with those studies that indicate that there is no significant correlation between gender and programming course outcomes (Ainsworth & VanLabeke, 2004; Carney & Levin, 2002; Werth, 1986).

2. Study 2: Combining schematic and realistic algorithm visualizations

As described above, the so-called transient information effect seems a plausible explanation of why animations are not so effective generally. An obvious response to this phenomenon would be to watch the animation repeatedly. It may seem that in such learning settings the overlap between the “select-process current information” and “remember-integrate previous

information” phases of learning process would not be so critical. In the case of algorithm animations/videos, for example, during the first viewing, learners might focus on how the steps of the algorithm are visualized and then, during the second viewing, they could try to comprehend the whole picture, the strategy that the algorithm applies.

On the other hand, there is evidence that simply repeating an information presentation (for example, repeatedly reading or listening to the same text) will not necessarily foster better learning (Fritz, Morris, Bjork, Gelman, & Wickens, 2000). The authors of the (Scheiter et al., 2009) study also found that simply studying materials twice that included both verbal and dynamic pictorial representations did not improve learning. However, these authors suggest that repeated viewing would improve meaningful learning if the second dynamic visualization stimulated students to engage in self-explanatory activities (Roy & Chi, 2005). In the present research students were required to comprehend the strategy of the studied algorithm, to realize a kind of whole picture phenomenon. In this situation the first viewing can generate such a feeling of lack (in those students who only fragmentally understood the algorithm) which then, during an immediate second viewing, could catalyze spontaneous self-explanatory activities.

To reduce redundancy, repeated viewing can be replaced by multiple external representations (MER). Ainsworth (2006) underlines that even isomorphic (informationally equivalent) representations can support complementary processes. Background cognitive processes behind learning with external representations are: understanding external representations, forming internal representations and integrating them into a mental model (Gilbert, 2008). Since different representations may complete each other, using multiple representations usually results in more accurate mental models of the domain (Rau, 2013). Supporting evidence in this sense, based on the Cognitive Theory of Multimedia Learning (Mayer, 2005) and Cognitive Load theory (Sweller, van Merriënboer, & Paas, 1998), mostly focus on the complementarities with respect to the sensory channels (auditory/visual) and/or the modalities of the representations (textual/pictorial or iconic/symbolic) (Ainsworth, 2006).

By contrast, learning with multiple graphical representations (MGR) implies the same symbol system. A possible side effect of this could be a cognitive overload in the pictorial part of working memory (Mayer, 2005). However, much prior research has shown that if they are used properly, MGR enhances students’ learning (Rau & Matthews, 2017). Key factors in this sense are: students are able to (1) accurately interpret each individual representation; (2) make connections among multiple representations (Ainsworth, 2006).

This two-step task is not straightforward. According to Rau, Michaelis and Fay (2015), while in “textual + graphical” settings the text could guide learners’ visual attention as they process the graphical representation (based on an assumable text processing fluency), in the case of “graphical + graphical” settings different graphical representations may share incidental perceptual features, which could lead to challenging connection making processes.

Another important question is how to schedule different graphical representations. The scheduling problem can be approached from the perspective the functional taxonomy of multiple representations too, proposed by Ainsworth (1999). She identifies three functions: complementary information and processes, constraining interpretation, deeper understanding. With respect to the constraining function Ainsworth and VanLabeke (2004) state that a familiar representation can support the interpretation of a less familiar one. Another way that constraint can be achieved is when the inherent properties of a representation make something explicit that is only implicit in the other representation. In the context of multiple dynamic representations both aspects suggest the following ordering: familiar/simple/explicit followed by less-familiar/complex/implicit. In other words, if two different dynamic visualizations follow one another, then the first one should contribute to making students more prepared for the interpretation of the second one.

Theses

In extension to the above literature review we report the following research results:

[T2.1] In case of repeatedly viewing the same visualization, HME enriched visualizations (dance videos) are more effective than schematic representation (abstract animations).

[T2.2] We assumed that watching the schematic visualization first would help interpreting the realistic one (cf. constraining function of MERs, Ainsworth, 2006) and, consequently, would contribute to better learning outcomes than the realistic+schematic condition.

Accordingly, we hypothesized that the realistic-last conditions will outperform the schematic-last

Details about the implemented experiment and the performed statistical analysis

The participants of this study consisted of 84 undergraduate students (14% females, which is a typical percentage in the case of science programmes of the university) from the Sapiientia University (at beginning of school year 2018-2019). In order to create four “statistically equivalent” groups we followed the below procedure: (1) students were sorted based on their

prior knowledge test (pre-test) results; (2) student- i ($i=1,2,\dots,84$) was assigned to group- j ($j=1,2,3,4$), where $j = (i-1) \text{ MOD } 4 + 1$. In other words: group-1 included students placed 1st, 5th, 9th, ...; group-2 included students placed 2nd, 6th, 10th, ...; group-3 included students placed 3rd, 7th, 11th, ...; group-4 included students placed 4th, 8th, 12th, ...

The pre-test evaluated students' prior knowledge with respect to selection-sort and bubble sort algorithms. Learning phase: As HME-realistic visualization we used the AlgoRythmics video of the shell-sort algorithm; In the role of schematic visualization a very expressive computer animation of this algorithm (New Media, 2013) was presented to students.

Evaluative phase: Participants were invited to answer 14 questions that aimed to reveal their comprehension of the shell-sort algorithm (post-test).

After a brief introduction, all the 84 participants took the prior knowledge test (pre-test) in an amphitheater. After the four groups had been formed they were invited into different classrooms. To each of the groups one of the following conditions was assigned: schematic-twice (group S-2, $n=20$), HME-realistic-twice (group R-2, $n=22$), schematic+ HME-realistic (group S-R, $n=21$), HME-realistic+schematic (group R-S, $n=21$). Group S-2 watched the schematic visualization twice (in succession). Group R-2 watched the HME enriched realistic visualization twice (in succession). Group S-R watched the schematic visualization first and then the realistic one. Group R-S watched the HME-realistic visualization first and then the schematic one. Each group watched the visualizations from the laptop of the teacher, using the video projector of the classroom.

We used ANOVA to check whether all conditions had comparable prior knowledge at the beginning of the experiment. The post-test scores were analyzed with ANCOVA. The independent variable was the instructional condition, and the dependent variable was the post-test result. Prior knowledge (pre-test scores) was used as a covariate. Results suggest that participants from the four conditions scored marginally different on the post-tests, $F(3,79)=2,61$, $p=0,057$, partial $\eta^2 = 0,09$. On the other hand prior knowledge was not significantly related to post-test scores ($F(1,79)=2,04$, $p=0,15$, partial $\eta^2 = 0,025$). Taking into account our hypotheses we performed further analyses (ANOVA, planned contrasts):

Schematic vs. Realistic. Comparing the performances of groups S-2 (HMErealistic-twice condition) and R-2 (schematic-twice condition) we found significant difference favoring the realistic group: $t(79)=2,24$, $p=0,027$.

Schematic+Realistic vs. Realistic+Schematic. Comparing the scores of groups S-R and R-S, no significant differences were detected: $t(79)=1,56$ ns.

Repeated vs. combined watching. Comparing participants' scores from groups S-2 and R-2 with the scores of their colleagues from groups S-R and R-S, no significant differences were detected: $t(79)=1,11$ ns.

Schematic-first vs. Realistic-first. Comparing participants' scores from groups S-2 and S-R with the scores of their colleagues from groups R-2 and R-S, no significant differences were detected: $t(79)=0,48$ ns.

Schematic-last vs. Realistic-last. Participants who watched the realistic visualization last performed significantly better than those ones who finished their learning session with the schematic visualization (groups S-2 and R-S vs. groups R-2 and S-R): $t(79)=2,69$, $p=0,008$.

We hypothesized (hypothesis-1) that visualizations involving human movement will overcome the schematic-only condition (because of the HME). The above presented results support this assumption only partially. Group R-S (despite the fact that the corresponding condition involved HME) attained the lowest score. On the other hand, group R-2 (HMErealistic-twice condition) performed significantly better than group S-2 (schematic-twice condition).

Our second hypothesis was that combined visualizations would support learning better than repeatedly watching the same visualization. The results of this study are consistent with the findings of the (Scheiter et al., 2009) and (Nugteren et al., 2014) studies. We also failed to show benefits associated with combining schematic and realistic representations of dynamic processes (apparently, contrary to what the literature on learning with MERs suggests). This suggests reciprocal advantages/disadvantages of both conditions.

We assumed, based on the constraining function of MERs (Ainsworth, 2006), that schematic+HMErealistic condition will outperform the HMErealistic+schematic condition (hypothesis-3). Although group S-R performed better than group R-S (65% vs. 57%) the difference was not statistically significant. This is in line with the results reported by the authors of the (Scheiter et al., 2009) study. It seems that the constraining function of MERs was not strong enough to balance the required processing resources for interpreting this specific realistic visualization and connecting it to the schematic one.

Since we assumed that deeper understanding of the algorithm will occur during the second viewing, another outcome-variant we predicted was that the realistic-last conditions will outperform the schematic-last ones (hypothesis-4). Results confirmed our expectation. A possible reformulation of this hypothesis is that to benefit maximally from this specific HMErealistic visualization, students need either a previous viewing of it or a previous watching of an isomorphic schematic presentation of it.

3. Study 3 Interactivity in AV environments

Theses

Considering these, in this study we focused on three different levels of interactivity on which we found both pro and contra opinions. In extension to the above literature review we report the following research results:

[T3.1] Interactive visualizations can benefit beginner (no prior programming knowledge) and intermediate (1-2 years of programming) students' results. Advanced (3-4 years of programming) students can understand the main concepts of the algorithm, regardless of the type of representation.

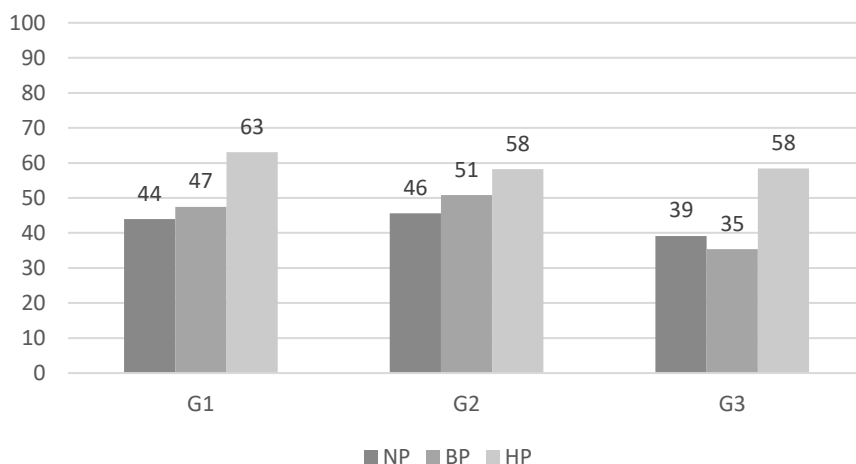
[T3.2] There is no significant difference between boys and girls.

[T3.3] There is a similarity between the learning style and question style. Full interactivity (C group) is a disadvantage in case of complexity questions.

Details about the implemented experiment and the performed statistical analysis

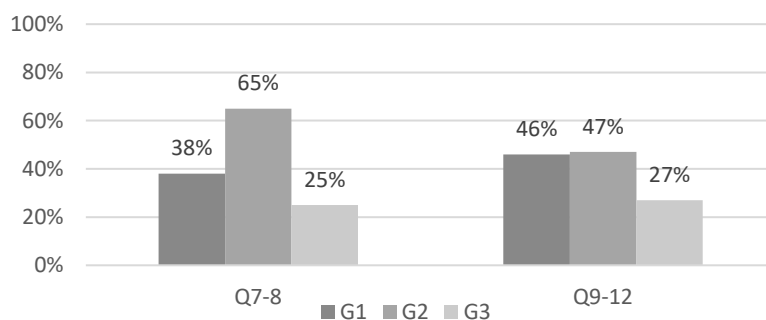
This study contains three phase experiment: *pre-test, learning phase and post-test*. The participants of this study consisted of 137 (14% girls) undergraduate students of 2019/2020 academic year. We identified 3 categories in which students were enrolled according to their prior programming experience: No-prior knowledge (NP: 0 years of high school programming experience) students, Basic-prior knowledge (BP: 1 or 2 years of high school programming experience) students and High-prior knowledge (HP: 3 or 4 years of high school programming experience) students. Students from each category were randomly assigned to the three groups G1 (0 interactivity), G2 (1/2 interactivity) and G3 (1 interactivity).

Results by prior programming knowledge and the level of interactivity – Post-test



We conclude that in case of beginner and intermediate students less interactivity can lead to better results, or at least to a consistent growth. It can be also observed, that full interactivity eventuates lower performance for all type of students. While HP students did their best in case of G1 group, both NP and BP students reached their maximum results in case of G2 group. Although these results are contrary to previous studies, which mentioned that interactivity is better to use in case of advanced students, it can be seen that both NP and BP students from G2 group performed well, which means that half interactivity was for their benefit. In contrary to our previous result, this emphasized that despite of the prior programming knowledge, with this learning opportunity, beginner students can also perform pretty well.

Q7-8 and Q9-12 by course type – BP students



In case of Q7-8 questions students from both G1 and G2 performed better, and in case of Q9-12 questions significantly better than their colleagues from G3 group. This also supports our fourth hypothesis, which declared that there is a similarity between the learning style and question style.

4. Study 4: The role of tutorial question asking in algorithm visualization environments

Previous research on the AlgoRhythmics environment concentrate mostly on the potential the dance choreographies (supplemented with animations) incorporate to support different categories of students in understanding the strategy of the algorithms (Katai, 2014a, 2014b, 2014c, 2015). Other characteristics of these studies are that the authors analyze only self-paced learning sessions and the examined learning environments supplement the viewing of the dance choreographies (and the attached animations) with the interactive orchestration of the algorithms. For example, the (Katai, 2015) study focuses on science versus humanities oriented undergraduate students and concludes that active involvement plays a crucial role in supporting the humanities-oriented students in assimilating the strategy of the studied algorithm. In addition, the above referred studies do not investigate if students are able to build on the knowledge they have acquired.

In this study we analyze a learning setting built around AlgoRhythmics videos where the principle of active involvement is implemented by question asking (with and without teacher guidance). Beside prior research in the field, our experience with these videos (as CS teachers) also inspired us, since we have observed that without teacher support the potential these choreographies incorporate, as CT promoter tools, is only partially exploited. For example, definitions often present the concept of algorithm efficiency as an important component of CT (Shute, Sun, & Asbell-Clarke, 2017; CSTE, 2017) and this concept is based on the best and worst case behaviour of algorithms. We have proposed to investigate the following question: are these visualization (created for given inputs) expressive enough to support students without prior knowledge in computing to imagine the best and worst case behaviour of algorithms? The attached questions, beside supporting and guiding students' thinking process, also help them focus on the domain-relevant aspects of the visualizations.

Asking questions to support learning with visual representations

Besides its evident advantages, Guzdial (2010) draws attention to possible negative side effects of contextualization. For example, overemphasizing a context can obstruct knowledge transfer. Although the AlgoRythmics environment involves the context only at the level of appealing decorative elements, the dangers of distraction still persist. Prior research shows that the effectiveness of learning with visual representations critically depends on students' ability to make sense of these educational materials (Wu & Rau, 2018). Several studies conclude that effective instructional activities could support students in this sense (Ainsworth, 2006; Rau, 2016). Without this support students often focus on irrelevant surface features and fail to depict domain-relevant concepts (especially in the case of decorative illustrations) (Wu & Rau, 2018).

Tutorial question asking could prove to be an efficient method of engaging students in meaningful learning (Graesser & Person, 1994). Boyer et al. (2010) reports on the positive results of asking effective questions to support students' learning in a CS educational context. These authors emphasise that the effective use of questions has the potential to prompt students to engage in valuable learning behaviours that they might not otherwise have undertaken. For example, it facilitates comprehension, encourages self-explanation, and reveals incomplete or incorrect knowledge. According to Tofade, Elsner and Haines (2013) using questions is an effective way to stimulate the recall of prior knowledge, promote comprehension, and build critical-thinking skills. Well worded questions can stimulate students to think about a topic in a new way. Effective teachers are able to formulate questions to fit the cognitive level of students. Other characteristics of the effective use of questions are: careful phrasing and word clarity; creating a psychologically safe environment; appropriate sequencing and balance; properly calibrated wait time.

Accordingly, we sequenced the algorithms according to the principle of moderate-progressive challenge: linear search, binary search, bubble sort (although bubble sort is not the most intuitive sorting strategy, after being visualized, students usually realize its strategy quite easily). The questions attached to each algorithm were also sequenced according to this principle.

Theses

In extension to the above literature review we report the following research results:

[T4.1] Blended learning has an important role in order to promote the development of algorithm visualization environments.

Details about the implemented experiment and the performed statistical analysis

We designed the following three phase learning-testing session:

1. After a brief introduction participant were invited to watch and analyze the dynamic visualizations of three algorithms (linear and binary search: dance video played twice; bubble sort: dance video plus animation); After each visualization they had to answer algorithm complexity questions;

(The visualizations helped students in realizing the strategies the algorithms apply; The attached question-sequences guided them in imagining the best and worst case behaviour of the studied algorithms);

2. By questioning (using additional leading questions) the instructor helped students realize/discover the right answers to the phase-1 questions;

3. Students were presented with the phase-1 learning conditions again, but built around a new algorithm (selection sort: dance video plus animation followed by algorithm complexity questions).

We tested the effectiveness of this three-phase learning unit on first year engineering students without any prior programming experience (experimental group). We anticipated that

- (Hypothesis-1) participants would assimilate the algorithms at a satisfactory level, even during the first phase of the learning session (as reflected in students' answers to the phase-1 questions);
- (Hypothesis-2) phase-2 instructional intervention will contribute significantly to students' understanding (as reflected in their answers to the phase-3 questions).

In order to have terms for comparison (with respect to Hypothesis-1) we involved, in our phase-1 analysis, those first year students too (control group) who were already initiated on the algorithms in question during their high school studies (according to their high school curriculum they studied the strategies of the algorithms without any algorithm complexity analysis). We expected that: (1) comparing phase-1 answers of the two groups would show no significant differences (the environment will support experimental group students to catch

up with their control group counterparts); (2) The experimental group's phase-3 performance would be significantly superior to their phase-1 performance (due to the phase-2 instructional intervention).

181 first year undergraduate students (male: 87%) were involved in the experiment. Students were grouped according to the number of years they had learned programming in high school: group-0 (0 year, 27%); group-1/2 (1 or 2 years, 25%); group-4 (4 years, 48%). The high school curriculum for group-1/2 included searching and quadratic sorting algorithms at a basic level. Accordingly, these participants had already been introduced with the strategies these algorithms apply, but time complexity related concepts (like best and worst case behaviour) were new to them. The group-4 students had studied both the algorithms and concepts of complexity. For the first phase of the experiment group-0 was the experimental group and group-1/2 the control. We were wondering if the visual learning environment we created had the potential to help group-0 students catch up with their group-1/2 counterparts. (Group-4 was involved in this analysis as a kind of secondary control group).

Our analysis included the phase-1 answers of all the three groups, and comparison of the corresponding phase-1 versus phase-3 answers of group-0. Interestingly, although students from group-1/2 had studied all the three algorithms previously, they did not outperform group-0 consequently. No significant differences were detected regarding the linear search and bubble sort algorithms and the synthesis question. Moreover, in the case of bubble sort, group-0 even outperformed group-1/2. The only significant difference (favouring group-1/2) was detected with respect to the binary search algorithm, the least intuitive one. This result partially supports our first hypothesis that the analysed learning environment has the potential to initiate non-majors (without prior knowledge in programming) in computer algorithms.

On the other hand, the fact that group-0 scored significantly lower than group-1/2 on the questions attached to the least intuitive algorithm (binary search), and that both group-0 and group-1/2 performed poorly relative to group-4, support our second expectation that without teacher guidance the potential this environment incorporates can be explored only partially. Comparing group-0 students' phase-1 and phase-3 results strengthened us in this observation. 57% of students realized that $N(N-1)/2$ swapping operations are needed for selection sort if the input sequence is sorted in descending order (worst case). This is a significantly better result (t-test: $p=0,05$) than in the case of the corresponding phase-1 question (regarding the bubble sort algorithm).

Phase-3 results emphasize the importance of the phase-2 discussion built around the phase-1 questions. Since the phase-3 learning environment was similar to the phase-1 setting, it is plausible to assume that phase-2 discussion contributed considerably to students' phase-3 performance. In addition, they performed better on the phase-3 algorithm, although the tutorial question-asking from phase-2 was limited to the phase-1 algorithms. The fact that students were able to extend their phase-1 comprehension to a new algorithm suggests that phase-2 discussion helped students see beyond the specific examples. Accordingly, phase-3 results confirm the hypothesis that the AlgoRhythmics environment, if complemented with tutorial question-asking, could be an effective instrument in introducing students with no prior knowledge, even with deeper CS concepts such as algorithm efficiency.

5. Study 5: Investigating young school students' computational thinking ability across grade levels

Over the last decade, continuous efforts have been made to bring computational thinking (CT) closer to K-12 education and even to K-9 education. Two complementary implementational approaches for the "CT for all" initiative is: (1) introducing new computing courses, and (2) infusing CT into the existing courses (Román-González, Pérez-González, Moreno-León, & Robles, 2018). For example, since 2014, Computer Science (CS) has been mandatory for pupils in the UK from age five upwards (Brown, Sentance, Crick, & Humphreys, 2014) and a growing recognition of the importance of CS education is observable in other countries too (European Schoolnet, 2015). In addition, Mannila et al. (2014) report on a survey distributed to K-9 teachers, aiming at revealing to what extent different aspects of CT are already part of the current curricula in various European countries and the U.S.

These focused endeavors implicitly suggest that current curricula do not contribute sufficiently to the development of learners' CT. Several studies emphasize that computing students lack a variety of skills that programming ability would require (Ahadi, Lister, Lal, Leinonen, & Hellas, 2017; Evans & Simkin, 1989; Simon, Chen, Lewandowski, McCartney, & Sanders, 2006). On the other hand, since CT is a combined skill with cross-disciplinary implications (Feaster, Ali, Zhai, & Hallstrom, 2014), one might conclude that even without an explicit focus on CS education students' CT might develop latently as they advance with the current curriculum. For example, Lewandowski and his colleagues report on a series of computing projects with the goal of identifying the commonsense knowledge beginners bring to the study of CS (Lewandowski et al., 2010). While the majority of previous research

focuses on assessing the CT level of certain age groups, the present study was motivated by the following basic question: Is there any detectable incidental progress in students' CT during their K-9 education? Instead of using one of the general CT tests proposed by previous work in the field (Román-González et al., 2018; Shute, Sun, & Asbell-Clarke, 2017), we chose to follow the same approach the authors of the above referred computing project applied. More explicitly, we have focused on testing whether there are differences in how 3rd, 5th, 7th and 9th grade learners (without any explicit prior experience with CT) relate to a specific learning task that assumes a certain level of CT.

Considering that algorithms are central to CT (Denning, 2009), the learning environment we created confronts students with “masked computer algorithms”. Learners are asked to identify with a dancer who is looking for his match in a dancer sequence. We took into account Denning's (2017) recent reminder that algorithms must control some computational model and step sequences that require human judgment should not be considered algorithms in the context of CT. In line with this, Katai (2014) states that algorithm visualization systems showing information that has extra meanings for learners may obstruct them in following strict computer algorithms. Accordingly, the members of the dancer sequence wear the numbers they represent on their back and the boy has to ask girls for a dance to realize who is his match (the girl dancing the same choreography and wearing the same number). In order to be able to perceive the differences in students' CT, the tasks included in the learning environment assume different levels of abstracting (Román-González et al., 2018).

In order to obtain an instrument accessible for 3rd grade students too, the testing tool we attached to the learning environment was built on the linear search strategy. We attempted to investigate CT ability by measuring to what extent students intuit the linear search algorithm before viewing the AlgoRhythmics video and how well they could assimilate the algorithm they were introduced to. This algorithm includes the majority of those algorithmic elements the CSTA standard suggests with respect to the 3-5 grade interval (CSTA, 2017).

We also took into account the definition Shute et al. (2017) gave to CT: “the conceptual foundation required to solve problems effectively and efficiently (i.e., algorithmically, with or without the assistance of computers) with solutions that are reusable in different contexts”. They included in their CT model the following facets: decomposition, abstraction, algorithms, debugging, iteration, and generalization. The assignments our testing instrument incorporates aim to reveal these components. Accordingly, we assumed that students' correct answers to the tasks reflect their CT ability since these required them to decompose the

algorithmic searching process into a succession of dances (representing comparisons) and to iterate through the girl sequence until the boy meets his match. With respect to the effectiveness and efficiency factor, some tasks are built around the concept of algorithm complexity. These assignments assumed quite advanced abstraction since students had to imagine and generalize the best- and worst-case behaviors of an algorithm that was presented to them, as a dance choreography, only for a single specific input.

Theses

In extension to the above literature review we report the following research results:

[T5.1] The rate and pace of CT growth depend on the nature of the current curriculum (Arts vs. Theoretical schools).

[T5.2] We propose a new testing tool (built around the linear search dance choreography that was recently added to the AlgoRythmics collection) for providing some measure of the CT skills of young school students. We have examined the pace at which some CT skills of young students (girls vs. boys from Art vs. Theoretical schools) latently develop as they advance with their current CS free curriculum. We have investigated the extent to which students of different ages can assimilate a basic computer algorithm (linear search).

Details about the implemented experiment and the performed statistical analysis

The main research question we addressed is: are there any detectable differences in how 3rd, 5th, 7th and 9th grade learners (without any explicit prior experience with CT) relate to learning tasks that assume a certain level of CT? More specifically:

- What is the pace of the potential CT growth? (RQ-1)
- Does the rate and pace of CT growth depend on the nature of the current curriculum (Arts vs. Theoretical schools)? (RQ-2)
- To what extent can students of different grade levels assimilate a basic computer algorithm (linear search) (RQ-3)
- To what extent are there signs of advanced CT at different grade levels? (RQ-4)
- Does the rate and pace of CT growth depend on the gender of the learners? (RQ-5)

The participants of this study consisted of 214 students (56% girls) from two public schools with a good reputation in their region. Since at the elementary level there are only theoretical (T) and art (A) specificities, the involved schools have these orientations (School-T and

School-A). Eight classes of grades 3, 5, 7, 9 were invited to participate in the experiment, at each level one class from both schools: classes 3T, 3A (music), 5T, 5A (fine-arts), 7T, 7A (fine-arts), 9T, 9A (fine-arts).

The testing process included six consecutive tasks (1, 2.1, 2.2.1, 2.2.2, 2.3.1, 2.3.2). Some of them referred to pictures presenting the girl sequence from behind and the boy with his number on his chest. Using this method, we achieved that the students see both the number the boy represents and the sequence the girls personify, but they are also able to perceive that the girls' numbers are invisible to the boy. Other tasks required students to imagine the best and worst case behaviour of the algorithm by providing number sequences that assume a minimal and maximal number of dances for the boy to find his match.

We performed the experiment during the first semester of the 2018/19 academic year. After a brief introduction participant were invited to login to an online testing application (Socrative) that included the items from Table 1. The testing sessions for all groups lasted about 30-35 minutes. In the remaining 10-15 minutes, we asked the students to interpret their answers (verbal probing). Task-1 aimed to reveal to what extent students intuit the linear search algorithm before viewing the AlgoRhythmics video. The role of Task-2 was to reveal how deeply students had assimilated the algorithm they were introduced to (by watching the video). An important detail of Task-2 was that students had no possibility to indicate at which end of the list their searching started.

Table 1. Questions and tasks the investigation procedure included.

Question-1	What is your gender? (male/female)	
Question-2	How much do you like informatics? (1..10)	
Question-3	How much do you love music and dance? (1..10)	
Task-1	Students were shown the picture, and they were asked to indicate the number of dances required for the boy to identify his match. It was emphasized that he does not see the numbers the girls are wearing and had to invite each girl for a dance to find out if she wears his number.	
Video	Subjects were invited to watch the flamenco dance choreography illustrating the linear search algorithm. Since the algorithm unfolds quite slowly (because of the choreography) the video was played at double speed. They were asked to focus on the strategy the boy from the video applies.	
Question-4	How much did you understand (in your opinion) of the boy's method? (1..10)	
Task-2	2.1	Students were shown a picture with a girl sequence of length 10 (the girl wearing the boys number was the 7th or 4th depending on the position of the head of the list). They were asked again to indicate the number of dances required for the boy to identify his match (it was emphasized that they had to implement the same method the boy had applied).

	2.2	Students were shown the picture and they were asked to give two number sequences of length 5 that require the minimal number (<i>subtask 2.2.1</i>) and the maximal number (<i>subtask 2.2.2</i>) of dances for the boy to find his match.
	2.3	Students were requested to generalize the algorithm for a girl sequence of length x by giving the required number of dances for the best case (<i>subtask 2.3.1</i>) and worst case (<i>subtask 2.3.2</i>) scenarios.

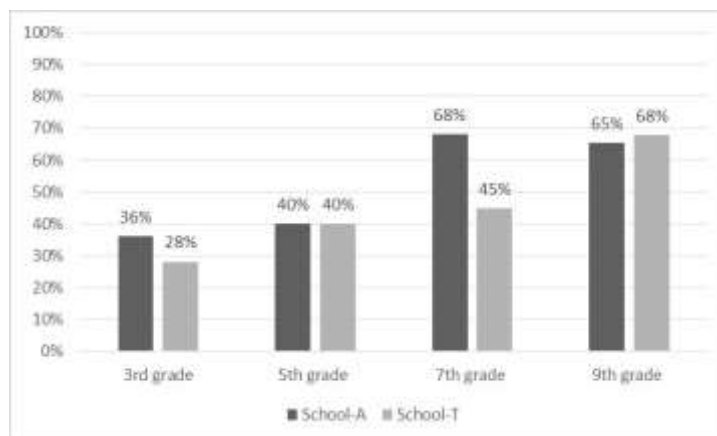
Table 2. All students' Task-1 and Task-2 performance percentages (means and standard deviations) grouped by grade level.

	Task-1 M	Task-2 (2.1, 2.2.1, 2.2.2, 2.3.1, 2.3.2) M(SD)	Tasks 2.1, 2.2 M(SD)
Grade 3	0.32	0.65 (0.18)	0.83 (0.22)
Grade 5	0.43	0.80 (0.17)	0.84 (0.18)
Grade 7	0.56	0.80 (0.24)	0.85 (0.23)
Grade 9	0.67	0.79 (0.20)	0.89 (0.19)

Considering Task-1 (to what extent students intuit the linear search algorithm) scores of all students of grades 3, 5, 7 and 9 from both schools (see Table 2) a gradual, but moderate growth can be noticed. While differences associated with the two-year shifts (3-5, 5-7, 7-9) were not significant, the four-year shifts resulted in significant increases (Fisher's exact test: shift 3-7, $p=0,01<0,05$; shift 5-9, $p=0,01<0,05$). To answer RQ-1 we can state that results suggest that the implicit contribution of the current K-9 curriculum to students' CT is relatively low since significant increase could only be detected with respect to four years shifts.

Investigating the two schools separately we found that, overall, students enrolled in art programmes performed slightly better than their counterparts from theoretical school (School-A: 53%, School-T: 46%; Fisher's exact test: $p=0,33>0,05$). On the other hand, the stages of growth differed significantly (see Figure).

The number of years each class has been studying in the corresponding school: School-A (black), School-T (white), other schools (grey).



The most relevant comparisons from the perspective of the “Art versus Theoretical programme analysis” are between classes 3A and 3T, 7A and 7T and the growth from grade 3A to grade 7A vs. the growth from 3T to 7T. Focusing on these pairs, we noticed that while the growth from grade 3 to 7 was significant (32%) in School-A (Fisher’s exact test: $p=0,04<0,05$) (mainly due to the increase from grade 5 to 7, 28%), in School-T it was relatively small (17%). In addition, it can also be observed that Class 3A performed considerably better than Class 3T (36% vs. 28%) and Class 7A performed substantially better (68% vs. 45%; Fisher’s exact test: $p=0,1$) than Class 7T. These differences reveal that young school students’ progress in CT could be affected by factors such as the educational programme and also suggest that art education may have extra potential to contribute to students’ CT. This finding reminded us of those successful initiatives that combines arts and CS education.

Students’ Task-2 performance can also be considered as CT measures since the role of these assignments was to reveal to what extent they were able to extract the linear search algorithm from the presented dance choreography. The Task-2 results (see Table 2) were analyzed with ANOVA. Results suggest significant differences: $F(210,3)=7,79$; $p=0,000<0,05$; partial $\eta^2 = 0,1$. As it can be discerned from Table 2, 3rd grade students (65%) were lagging behind their

5th, 7th and 9th grades colleagues who performed equally well (80%, 80%, 79%). Analyzing the responses of 3rd grade students, we noticed that they performed significantly lower than their higher-grade mates in the case of Task-2.3.2 (only 12% of them answered x). This is plausible since denoting an arbitrary value by a letter (for example x) is not an accessible concept yet for this age level (Kramer, 2007).

After we had eliminated Task-2.3 scores from our analysis we repeated the ANOVA test for students' Task-2.1 and Task-2.2 performances. We were surprised to find no significant differences among the four grade levels ($F(210,3)=0,83$; $p=0,47>0,5$). These excellent Task-2 results (including the fact that lower grade learners managed to catch up with their higher-grade counterparts) are in line with a considerable number of existing experiences demonstrating that it is indeed possible to teach CS-based concepts successfully in primary and secondary education (Gander et al., 2013).

As a next step, we compared students' performance based on their gender. We found that male and female participants performed equally well in all tasks. For example, Task-1 scores of both male and female students was 49% and for Task-2 the corresponding results were 74% and 77%. These results are consistent with those studies that indicate that there is no significant correlation between gender and programming course outcomes (Ahadi et al., 2017; Byrne & Lyons, 2001; Werth, 1986). We also observed that despite the shoulder-to-shoulder performance, boys rated significantly higher on questions 2 (How much do you like informatics?) and 4 (How much did you understand the linear search method?) than girls. This is also in line with previous research that identified female learners' self-conceptions in computing (e.g. self-efficacy) and their perceived lack of confidence amongst colleagues (despite their obvious abilities and successes) as important factors which deter them from entering, and staying, in academic computing (Ahadi et al., 2017; Stout & Tamer, 2016). As we expected girls rated significantly higher on Question 3 (How much do you love music and dance?).

BIBLIOGRAPHY AND REFERENCES

- Ahadi, A., Lister, R., Lal, S., Leinonen, J., & Hellas, A. (2017, January). Performance and Consistency in Learning to Program. In *Proceedings of the Nineteenth Australasian Computing Education Conference* (pp. 11-16). ACM.
- Aho, A. V. (2011). Computation and computational thinking. *Ubiquity Symposium*. DOI: 10.1145/1895419.1922682
- Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, 55(7), 832–835.
- Ainsworth, S., & VanLabeke, N. (2004). Multiple forms of dynamic representation. *Learning and instruction*, 14(3), 241-255.
- AlgoRhythmicsenvironment. (2020). <https://algorhythmics.ms.sapientia.ro/>.
- Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661-670.
- Ayres, P., & Paas, F. (2007). Making instructional animations more effective: A cognitive load approach. *Applied Cognitive Psychology*, 21, 695–700.
- Berney, S., & Betrancourt, M. (2016). Does animation enhance learning? A meta-analysis. *Computers & Education*, 101, 150–167.
- Butcher, K. R. (2006). Learning from text with diagrams: Promoting mental model development and inference generation. *Journal of Educational Psychology*, 98(1), 182.
- Byrne, M. D., Catrambone, R., & Stasko, J. T. (1996). Do algorithm animations aid learning? Georgia Institute of Technology.
- Byrne, P., & Lyons, G. (2001, June). The effect of student attributes on success in programming. In *ACM SIGCSE Bulletin* (Vol. 33, No. 3, pp. 49-52). ACM.
- Carney, R. N., & Levin, J. R. (2002). Pictorial illustrations still improve students' learning from text. *Educational psychology review*, 14(1), 5-26.
- Castro-Alonso, J. C., Ayres, P., & Paas, F. (2014). Dynamic visualisations and motor skills. In *Handbook of human centric visualization* (pp. 551-580). Springer, New York, NY.
- Castro-Alonso, J. C., Ayres, P., Wong, M., & Paas, F. (2018). Learning symbols from permanent and transient visual presentations: Don't overlay the hand. *Computers & Education*, 116, 1–13.
- Comenius (1992): *Pampedia*. (IV.rész, ford. Bollók János), B-A-Z Levéltár, Sárospatak. IX. fejezet, A kisgyermekkor iskolája, az anyaöl, vagyis az emberi sarjnak a születéstől körülbelül hatéves koráig tartó gondos nevelése. 68-80.
- de Koning, B. B., & Tabbers, H. K. (2011). Facilitating understanding of movements in dynamic visualizations: An embodied perspective. *Educational Psychology Review*, 23(4), 501–521.
- Denning, P. J. (2009). The profession of IT Beyond computational thinking. *Communications of the ACM*, 52(6), 28-30.
- Denning, P. J. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, 60(6), 33-39.
- Evans, G. E., & Simkin, M. G. (1989). What best predicts computer proficiency? *Communications of the ACM*, 32(11), 1322-1327.
- Feaster, Y., Ali, F., Zhai, J., & Hallstrom, J. O. (2014, June). Serious toys: three years of teaching computer science concepts in K-12 classrooms. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 69-74). ACM.
- Fritz, C. O., Morris, P. E., Bjork, R. A., Gelman, R., & Wickens, T. D. (2000). When further learning fails: Stability and change following repeated presentation of text. *British Journal of Psychology*, 91(4), 493-511.
- Goldstone, R. L., & Son, J. Y. (2005). The transfer of scientific principles using concrete and idealized simulations. *The Journal of the Learning Sciences*, 14(1), 69-110.
- Graesser, A. C., & Person, N. K. (1994). Question asking during tutoring. *American educational research journal*, 31(1), 104-137.

- Guzdial, M., & Tew, A. E. (2006, September). Imagineering inauthentic legitimate peripheral participation: an instructional design approach for motivating computing education. In *Proceedings of the second international workshop on Computing education research* (pp. 51-58). ACM.
- Höffler, T. N., & Leutner, D. (2007). Instructional animation versus static pictures: A meta-analysis. *Learning and Instruction*, 17(6), 722-738.
- Höffler, T. N., Leutner, D. (2007). Instructional animation versus static pictures: A meta-analysis. *Learning and instruction*, 17(6), 722-738.
- Jarc, D. J., Feldman, M. B., & Heller, R. S. (2000). Assessing the benefits of interactive prediction using web-based algorithm animation courseware. *ACM SIGCSE Bulletin*, 32(1), 377-381.
- Katai, Z. (2014, June). Selective hiding for improved algorithmic visualization. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 33-38). ACM.
- Káta, Z. (2015). The challenge of promoting algorithmic thinking of both sciences-and humanities-oriented learners. *Journal of Computer Assisted Learning*, 31(4), 287-299.
- Katai, Z., Toth, L., & Adorjani, A. K. (2014). Multi-Sensory Informatics Education. *Informatics in Education*, 13(2), 225-240.
- Káta, Z., Juhász, K., & Adorjáni, A. K. (2008). On the role of senses in education. *Computers & Education*, 51(4), 1707-1717.
- Korhonen, A., & Malmi, L. (2000). Algorithm simulation with automatic assessment. *ACM SIGCSE Bulletin*, 32(3), 160-163.
- Lau, W. W., & Yuen, A. H. (2009). Exploring the effects of gender and learning styles on computer programming performance: implications for programming pedagogy. *British Journal of Educational Technology*, 40(4), 696-712.
- Lewandowski, G., Bouvier, D. J., Chen, T. Y., McCartney, R., Sanders, K., Simon, B., & VanDeGrift, T. (2010). Commonsense understanding of concurrency: computing students and concert tickets. *Communications of the ACM*, 53(7), 60-70.
- Lin, L., & Li, M. (2018). Optimizing learning from animation: Examining the impact of biofeedback. *Learning and Instruction*, 55, 32-40.
- Lowe, R. K. (1999). Extracting information from an animation during complex visual learning. *European Journal of Psychology of Education*, 14, 225-244.
- Lowe, R. K., & Boucheix, J.-M. (2017). A composition approach to design of educational animations. In R. Lowe, & R. Ploetzner (Eds.). *Learning from dynamic Visualizations: Innovations in research and application* (pp. 5-30). Berlin: Springer.
- Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014, June). Computational thinking in K-9 education. In *Proceedings of the working group reports of the 2014 on innovation & technology in computer science education conference* (pp. 1-29). ACM.
- Mayer, R. E. (2005). Cognitive theory of multimedia learning. In R. E. Mayer (Ed), *The Cambridge handbook of multimedia learning* (pp. 31-48). Cambridge University Press.
- Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist*, 38(1), 43-52.
- McCartney, R., Bouvier, D. J., Chen, T. Y., Lewandowski, G., Sanders, K., Simon, B., & VanDeGrift, T. (2009, August). Commonsense computing (episode 5): algorithm efficiency and balloon testing. In *Proceedings of the fifth international workshop on Computing education research workshop* (pp. 51-62). ACM.
- Moreno, R., Ozogul, G., & Reisslein, M. (2011). Teaching with concrete and abstract visual representations: Effects on students' problem solving, problem representations, and learning perceptions. *Journal of Educational Psychology*, 103, 32-47.
- Naps, T. L., Eagan, J. R., & Norton, L. L. (2000, May). JHAVÉ—an environment to actively engage students in Web-based algorithm visualizations. In *ACM SIGCSE Bulletin* (Vol. 32, No. 1, pp. 109-113). ACM.
- Nugteren, M. L., Tabbers, H. K., Scheiter, K., & Paas, F. (2014). Simultaneous and Sequential Presentation of Realistic and Schematic Instructional Dynamic Visualizations. In *Handbook of human centric visualization* (pp. 605-622). Springer, New York, NY.

- Paas, F., & Sweller, J. (2012). An evolutionary upgrade of cognitive load theory: Using the human motor system and collaboration to support the learning of complex cognitive tasks. *Educational Psychology Review*, 24(1), 27-45.
- Paas, F., & Sweller, J. (2012). An evolutionary upgrade of cognitive load theory: Using the human motor system and collaboration to support the learning of complex cognitive tasks. *Educational Psychology Review*, 24(1), 27-45.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc..
- Pukánszky Béla, N. (n.d.). *Neveléstörténet*. Bp., Nemzeti Tankvk., 1997. 185. o
- Rau, M. A. (2013). *Conceptual learning with multiple graphical representations: Intelligent tutoring systems support for sense-making and fluency-building processes* (Unpublished doctoral dissertation). Carnegie Mellon University, Pittsburgh, PA.
- Rau, M. A. (2017). Conditions for the effectiveness of multiple visual representations in enhancing STEM learning. *Educational Psychology Review*, 29(4), 717-761.
- Rau, M. A., & Matthews, P. G. (2017). How to make 'more' better? Principles for effective use of multiple representations to enhance students' learning about fractions. *ZDM*, 49(4), 531-544.
- Rau, M. A., Aleven, V., & Rummel, N. (2013). Interleaved practice in multi-dimensional learning tasks: Which dimension should we interleave?. *Learning and Instruction*, 23, 98-114.
- Rau, M. A., Michaelis, J. E., & Fay, N. (2015). Connection making between multiple graphical representations: A multi-methods approach for domain-specific grounding of an intelligent tutoring system for chemistry. *Computers & Education*, 82, 460-485.
- Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior*, 72, 678-691.
- Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2018). Can computational talent be detected? Predictive validity of the Computational Thinking Test. *International Journal of Child-Computer Interaction*, 18, 47-58.
- Roy, M., & Chi, M. T. (2005). The self-explanation principle in multimedia learning. *The Cambridge handbook of multimedia learning*, 271-286.
- Scheiter, K., Gerjets, P., Huk, T., Imhof, B., & Kammerer, Y. (2009). The effects of realism in learning with dynamic visualizations. *Learning and Instruction*, 19(6), 481-494.
- Settle, A., Goldberg, D. S., & Barr, V. (2013, July). Beyond computer science: computational thinking across disciplines. In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (pp. 311-312). ACM.
- Shaffer, C. A., Cooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S., & Edwards, S. H. (2010). Algorithm visualization: The state of the field. *ACM Transactions on Computing Education (TOCE)*, 10(3), 9.
- Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22, 142-158.
- Simon, B., Chen, T. Y., Lewandowski, G., McCartney, R., & Sanders, K. (2006, September). Commonsense computing: what students know before we teach (episode 1: sorting). In *Proceedings of the second international workshop on Computing education research* (pp. 29-40). ACM.
- Simon, B., Kinnunen, P., Porter, L., & Zazkis, D. (2010, June). Experience report: CS1 for majors with media computation. In *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education* (pp. 214-218). ACM.
- Stout, J., & Tamer, B. (2016, February). Collaborative learning eliminates the negative impact of gender stereotypes on women's self-concept. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education* (pp. 496-496). ACM.
- Sweller, J., Van Merriënboer, J. J., & Paas, F. G. (1998). Cognitive architecture and instructional design. *Educational psychology review*, 10(3), 251-296.
- Sweller, J., Clark, R., & Kirschner, P. (2010). Teaching general problem-solving skills is not a substitute for, or a viable addition to, teaching mathematics. *Notices of the American Mathematical Society*, 57(10), 1303-1304.

Tew, A. E., McCracken, W. M., & Guzdial, M. (2005, October). Impact of alternative introductory courses on programming concept understanding. In Proceedings of the first international workshop on Computing education research (pp. 25-35). ACM.

Tofade, T., Elsner, J., & Haines, S. T. (2013). Best practice strategies for effective use of questions as a teaching tool. *American journal of pharmaceutical education*, 77(7).

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, 59(236), 433.

Tversky, B., Morrison, J. B., & Betrancourt, M. (2002). Animation: can it facilitate?.

Werth, L. H. (1986). Predicting student performance in a beginning computer science class (Vol. 18, No. 1, pp. 138-143). ACM.

Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725.



Registry number:

DEENK/26/2021.PL

Subject:

PhD Publication List

Candidate: Erika Osztián

Doctoral School: Doctoral School of Mathematical and Computational Sciences

List of publications related to the dissertation

Foreign language scientific articles in international journals (1)

1. Kátai, Z., Osztián, E.: Improving AlgoRhythmic Teaching-Learning Environment by Asking Questions.
Int. J. Instr. 14 (2), 27-44, 2021. ISSN: 1694-609X.

Hungarian conference proceedings (4)

2. Osztián, E., Kátai, Z.: Az algoritmikus gondolkodás vizsgálata különböző korosztályú tanulóknál, e-learning tesztelési környezetben.
In: INFODIDACT 2019. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika az Oktatásért és az Információs Társadalomért Alapítvány, Budapest, 1-14, 2020. ISBN: 9786158060837
3. Kátai, Z., Osztián, E.: Algoritmikus gondolkodás fejlesztése különböző korosztályú és szakos tanulóknál.
In: ENELKO 2018 XIX. Nemzetközi Energetika-Elektrotechnika Konferencia, SzámOkt 2018 XXVIII. Nemzetközi Számítástechnika és Oktatás Konferencia. Szerk.: Biró Károly-Ágoston, Sebestyén-Pál György, Szabó Loránd, Erdélyi Magyar Műszaki Tudományos Társaság, Tusnádfürdő, 225-230, 2018, (ISSN 1842-4546)
4. Kátai, Z., Osztián, E., Vekov, G. K., Osztián, P. R.: Algo-Ritmika: új dimenziók.
In: INFODIDACT 2017. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika az Oktatásért és az Információs Társadalomért Alapítvány, Budapest, 1-11, 2017. ISBN: 9786158060813
5. Osztián, P. R., Kátai, Z., Osztián, E., Vekov, G. K.: Algoritmika és művészet, avagy kupacrendezés kalotaszegi módra = Algorithms and art Heapsort with traditional Transylvanian dance.
In: ENELKO 2017 XVIII. Nemzetközi Energetika-Elektrotechnika Konferencia, SzámOkt 2017 XXVII. Nemzetközi Számítástechnika és Oktatás Konferencia, Kolozsvár, 2017. október 12-15. Szerk.: Biró Károly-Ágoston, Sebestyén-Pál György, Erdélyi Magyar Műszaki Tudományos Társaság, Cluj-Napoca, 199-203, 2017, (ISSN 1842-4546)





Foreign language conference proceedings (4)

6. Kátai, Z., Osztián, E., Lőrincz, B.: The effect of multiple graphical representations on learning algorithms.
Turk. Online J. Educ. Technol. 2018 (2), 703-708, 2018. ISSN: 2148-7242.
7. Osztián, E., Kátai, Z., Vekov, G. K.: Multi-dimensional expansion of Algo-Rhythmics.
Turk. Online J. Educ. Technol. 2017 (3), 573-578, 2017. ISSN: 2148-7242.
8. Kátai, Z., Osztián, E., Vekov, G. K.: Promoting computational thinking by artistically enhanced algorithm visualization.
In: INFODIDACT 2016. Szerk.: Szlávi Péter, Zsakó László, Webdidaktika az Oktatásért és az Információs Társadalomért Alapítvány, Budapest, 1-12, 2016. ISBN: 9786158060806
9. Osztián, E.: Stimularea creativității tehnico - științifice la disciplinele Tehnologie informației și a comunicațiilor și Informatică prin programarea roboților Lego.
Simpoz. Nat. Edmond Nicolau. 9, 222-232, 2015. EISSN: 2069-9530.

Hungarian abstracts (1)

10. Kátai, Z., Osztián, E., Vekov, G. K.: Számítógépes gondolkodás fejlesztése vegyes oktatási környezetben.
In: XXIII. Multimedia in Education Conferences, Presa Universitară Clujeană, Cluj-Napoca, 104, 2017. ISBN: 9786063701832





List of other publications

Foreign language conference proceedings (2)

11. Osztian, P. R., Katali, Z., Osztian, E.: Algorithm Visualization Environments: degree of interactivity as an influence on student-learning.
In: 2020 IEEE Frontiers in Education Conference (FIE), IEEE, dPiscataway, 1-8, 2020, (ISSN 2377-634X) ISBN: 9781728189611
12. Nagy, E. J., Osztian, P. R., Cosma, C., Katali, Z., Osztian, E.: Looking for the Optimal Interactivity Level in the AlgoRhythmic Learning Environment.
In: EdMedia + Innovate Learning : 2019 conference. Ed.: Theo Bastiaens, Association for the Advancement of Computing in Education, Waynesville, 106-114, 2019. ISBN: 9781939797421

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

21 January, 2021

