

NYIRŐ PÉTER

A gyakorlati programozás és játékfejlesztés tanításának alternatív módszere grafikai felületek támogatásával

1. A PROGRAMOZÁS TANÍTÁSÁNAK MÓDSZERTANI MEGKÖZELÍTÉSE

A számítástechnika, valamint az infokommunikációs technológiák nagymértékű fejlődésének köszönhető, hogy társadalmunk információközpontúvá vált. (Siemens, 2005, 2006) Ennek hatására, ahogy sok más diszciplína is, úgy a pedagógia tudományterülete is átalakulóban van. (Benedek, 2007; Miskolczi, 2010) E változás a tanulás főszereplői számára is komoly kihívást jelent az oktatás legkülönbözőbb szintjein, így a felsőoktatásban is, ahol a diákoknak egyre nagyobb információmenyiséget kell elsajátítaniuk egyre rövidebb idő alatt. (Molnár, 2011; Molnár és Benedek, 2013) Több kutatás bizonyította, hogy az emberi gondolkodás elsősorban vizuális alapú, s az ember ezt az adottságát kihasználva lényegesen nagyobb hatékonysággal tud elméleti ismereteket memorizálni és tanulni. (Bleed, 2005; Nyíri, 2003; Visual Education, 2013) Ugyan sok helyen megjelenik már a „vizuális oktatás” gyakorlati alkalmazása, de jellemzően még ma is a ha-

gyományos szöveg- és könyvalapú formák a leginkább elfogadottak és alkalmazottak a pedagógiában. (Minden, ami Közgáz, 2010)

Az informatikatanítás – és ezen belül a gyakorlati programozás oktatásának az esetében ennek az ellentmondásnak egy szélsőséges változata tapasztalható. A számítógépek nagymértékű fejlődésének

köszönhetően ma már nem okoz problémát a képek, hangok, animációk és videók megjelenítése bármilyen elektronikai berendezésen. (Molnár, 2014) Ám a programozás oktatása sok helyen még mindig a szöveges formát támogatja, kihasználatlanul hagyva az informati-

a programozás oktatása sok helyen még mindig a szöveges formát támogatja, kihasználatlanul hagyva az informatikában és a médiában rejlő bőséges vizuális potenciált

kában és a médiában rejlő bőséges vizuális potenciált. Mindez azért vált jelentős méretű problémává napjainkra, mert az információközpontú társadalomban az élet minden területén elengedhetetlen az informatikai tudás, sok helyen pedig a programozás alkalmazása is ugyanilyen. Éppen ezért szükséges jelentős figyelmet fordítani a programozás oktatására és a lehetőségekhez mérten hatékonyabb módszert alkalmazni a jelenleginél.

1.1 A célok meghatározása

Célunk a programozásoktatás egy olyan – Magyarországon újszerűnek mondható – módszerének bemutatása és gyakorlatba ültetése, amely teljes mértékben kihasználja a számítógépek adta vizuális lehetőségeket. A koncepció az elméletorientált oktatás helyett a gyakorlatra és a megszerzett tudás *alkalmazhatóságára* koncentrálni elsősorban. (Katona és Kurdi, 2008) A résztvevői motiváció növelésének érdekében az általunk kidolgozott oktatási módszer elsősorban a számítógépes játékfejlesztésre, a játékkal való tanulásra koncentrálni. (Haden, 2006; Kirkland és O’Riordan; 2008) A játékfejlesztés további előnye, hogy a programozáshoz szorosan kapcsolódó multimédiás és általános számítógépes ismeretek mind megjelennek a képzés során – ezért általában komplexebb figyelmet követel, mint egy funkcionális program tervezése. A szórakoztatóipar mint célipar részére készülő komplex projektek az általános szoftverfejlesztéshez szükséges ismereteket és képességeket maradéktalanul tartalmazzák. (Chen és Cheng, 2007; Rajaravivarma, 2005) A módszertan sikerességének vizsgálata érdekében rendszeresen indulnak kurzusok 2013 szeptembere óta, amelyek elégedettségvizsgálatát röviden bemutatjuk a továbbiakban.

1.2 Az új oktatásmódszertan kidolgozásának előzményei

A mai népszerű programnyelvek (például: Java, C, C++, C#, Objective-C, PHP, Python, Ruby) mindegyike szöveges formát

használ. (Mikoluk, 2013) A szöveges forma előnye, hogy gyorsabban lehet vele dolgozni, mint a grafikus felületeket használó programnyelvekkel. Hátránya viszont, hogy egy adott programnyelv összes helyesírási formáját és elméleti felépítését meg kell tanulni, mielőtt azt széles körben alkalmazni lehetne. Egy kezdő programozó számára nehézséget okoz a szöveges programozás tanulása, mert az összes elméleti szabályt figyelembe kell vennie, mielőtt al-

a koncepció az elméletorientált oktatás helyett a gyakorlatra és a megszerzett tudás alkalmazhatóságára koncentrálni elsősorban

kalkulálni tudná a nyelvet. A szintaktika és a gondolkodásmód elsajátítása mellett egy-egy program speciális felépítésének tervezésére és az algoritmusok funkcionális alkalmazására már kevesebb figyelmet tud fordítani.

Ezen komplex feladatok elvégzéséhez elengedhetetlen a programozásban alkalmazott logikai matematika ismerete és alkalmazása.

A grafikus programnyelvek (például App Inventor, Scratch, LabVIEW) a szöveges nyelvekkel ellentétben egy kezdő programozó számára sokkal átláthatóbbak és könnyebben tanulhatóak. Ennek oka, hogy míg egy gyakorlott programozó a programsorok megírásakor sokszor fejben tartja a kódok kapcsolódási logikáját, a kezdőknek szükségük van annak vizuális megjelenítésére. A grafikus nyelvben írt program logikája a vizuális megjelenítés miatt könnyebben átlátható, továbbá szintaktikája is jóval egyszerűbb. Ebből következik, hogy egy grafikus programnyelv használatához kevesebb elméleti háttérre van szükség, a programtervezés során kevesebb figyelmet kell fordítani a helyesíráásra, könnyebb koncentrálni a program végső felépítésére. Ugyanakkor a grafikus programnyelvek felhasználásának keretei sokkal szűkebbek, alkalmazásuk pedig *hosszú távon* már nem

olyan gyors, mint egy *jól betanult* szöveges programnyelv.

Amíg tehát a kezdő programozóknak a grafikus programnyelvek tanulása a célvezetőbb, addig a tapasztaltabbak számára a szöveges forma az eredményesebb. A kettő között viszont sok programnyelv esetében *nincs átmenet*. A különböző fejlesztők által kiadott programozói szoftverek ilyen különbségei miatt a grafikus programnyelvből szövegesbe történő áttérés általában nehézséget okoz, megnehezíti a különböző alkalmazások között a tudás transzferálásának lehetőségét. (Leutenegger és Edgington, 2004)

Az általunk kidolgozott tanítási módszer alapötlete néhány számítógépes játék alapján fogalmazódott meg. A programozás játékon keresztül történő oktatása, főként a fiatal generációk esetében, a számítógéphasználat adott komplexitása miatt kézenfekvő módszernek ígérkezett. (Pap-Szigeti és Pásztor, 2008; Brain, 2009)

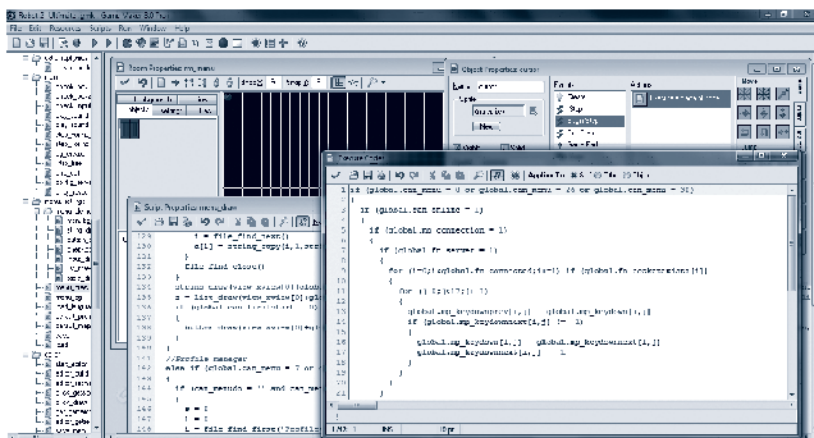
Több programot is írtak már a célból, hogy valamilyen vizuális-játékos módszerrel próbálják a programozás logikáját megmutatni és a tanulóval begyakoroltatni.

(Anderson és McLoughlin, 2007; Lockhart, 2013; Pinola, 2013; Romano, 2013; Finley, 2013, Heussner, 2013; Kurzweil, 2013; Primerlabs, 2013) Hatékonyság szempontjából valóban jó eredményt

mutat a legtöbb kezdeményezés, viszont a programokban található szűkített lehetőségek miatt ezek meg sem közelítik egy valódi programozási nyelv komplexitását. (Teach Kids to Program, 2013) Ezért ezek a játékok nem alkalmazhatóak széles körben – mindenképp egy komplexebb fejlesztői eszközre van szükség. Sajnos, csupán néhány fejlesztői program alkalmas oktatási célra – ezekben a grafikus és a szöveges nyelv megfelelő módon, fokozatos áttérést biztosítva jelenik meg. A kurzusainkon alkalmazott *Game Maker* program e ritka szoftverek közé tartozik (1. ábra).

1. ÁBRA

Játékefejlesztés a kurzusokon alkalmazott Game Maker 8.0 használatával



Elöl szöveges, hátul grafikus programozási felület látható.

FORRÁS: saját képernyőkép

1.3 Alkalmazott módszertan

A felsőoktatásban megjelenő programozói kurzusok szinte teljes köre a hagyományosnak mondható, elméletorientált oktatás-módszertant részesíti előnyben az ismeretek átadása során. Eszerint a diákoknak előbb meg kell érteniük a programnyelv szintaktikáját, a struktúrák és algoritmusok működését, hiszen csak ezután lesznek képesek mindezt alkalmazni. Ez a módszer az átfogó és részletes ismeretet ad az adott programnyelvről és annak használatáról, hátránya azonban, hogy az elméleti ismeretek elsajátítása a gyakorlati részt megelőzve nehezebb és emiatt lassabb folyamat. Mivel a megszerzett tudásnak a munka világában való alkalmazhatóságát főleg a gyakorlati feladatok határozzák meg, a rendelkezésre álló rövid idő alatt igazi kihívást jelent egy programozó kitanítása a fenti módon. (Tóth és Johanyák, 2007)

A hagyományos oktatási módszertannal szemben áll a gyakorlatorientált programozás, amely egy erre kialakított felületen keresztül, vizuális *hatásokkal* szemlélteti az egyes funkciókat és lehetővé teszi, hogy *alkalmazásokon keresztül* sajátítsák el a diákok ugyanazt az elméleti tudást. (Molnár, 2012) Így rövidebb idő alatt alkalmazhatósági szempontból nagyságrenddel nagyobb anyagot lehet megtanítani, továbbá a gyakorlatok alapján a későbbi elméleti magyarázatok is könnyebben értelmezhetők.

A programválasztásnál figyelembe kell venni a folytonosság elvét, melynek értelmében kerülni kell a grafikus és szöveges programozói felületek közötti *ugrászerű* átmenetet. Ezt egészíti ki a játékkfejlesztés pozitív hatásai iránti igényünk.

Összességében egy olyan fejlesztői eszközre van szükségünk, amelynek segítségével *a mai szoftverigényeknek megfelelő* számítógépes játékot lehet létrehozni. Ezen megfontolások alapján választottuk ki a kutatási munkához a Game Makert. Ennek az eszköznek a segítségével mind kezdő, mind haladó programozók begyakorolhatják a játékkfejlesztéshez is szükséges alapvető programozói feladatokat és megismerhetik a jellemző kihívásokat. A tanulás során nemcsak elméleti tudásra tesznek szert, hanem azt szinte azonnal alkalmazhatják is.

Az eszköz segítségével általános ismereteket szerezhetnek a C alapú objektumorientált programkódok elkészítéséről, az általános programtervezésről és a szükséges médiaismeretéről egyaránt.

A Game Maker névre hallgató, grafikus

felületet alkalmazó játékkfejlesztői eszközt egy holland informatikatanár, Markus Hendrik Overmars fejlesztette ki és mutatta be, első változatában 1999-ben. Célja az volt, hogy azok a gyerekek is képesek legyenek számítógépes játékokat készíteni, akik (még) nem tudnak programozni. (Overmars, 2004) Programnyelve a C nyelvekhez nagyon hasonló GML (Game Maker Language), amely kifejezetten úgy készült, hogy a grafikus programozói felületből a szövegesbe történő átállás *folyamatos, ugrásmentes* lehessen. A program ezt a különlegességét annak köszönheti, hogy a grafikus elemek akár *egyesével lecserélhetőek* szöveges formára, alkalmazkodva a diák tanulási folyamatához. A program újabb és újabb változatát a mai napig fejleszti egy angol cég, a Yoyo games. (Yoyo games, 2015) A társaság angol nyelvű weboldala teljes körű ismertetést nyújt a program tudásáról és használatáról (www.yoyogames.com).

a tanulás során nemcsak elméleti tudásra tesznek szert, hanem azt szinte azonnal alkalmazhatják is



A Game Maker 8.0 logója

2. A GYAKORLATORIENTÁLT OKTATÁSI MÓDSZER ALKALMAZÁSA A KÉPZÉSEKBEN

A kidolgozott oktatási módszert kurzusok keretei között lehet alkalmazni, bármely korosztálynál. Egy kurzus terjedelme fél vagy egy év lehet az iskolarendszerhez, illetve iskolaszervezethez igazodva. Felsőoktatásban vagy az iskolarendszeren kívüli oktatás alá tartozó felnőttképzésekben a kezdő kurzus fél évet, azaz heti egy, összesen tizenkét 1,5 órás előadást jelent azok számára, akik korábban még sosem programoztak. A kurzus végére azok, akik emellett elvégezték a megfelelő mennyiségű otthoni munkát is, képesek lesznek önállóan saját grafikus szoftvert írni asztali számítógépre.¹ Az elsajátított ismeret, a megszerzett tudás – s hogy milyen bonyolultságú programot képes írni a tanuló – csupán az önszorgalmon múlik. A haladó kurzus szintén heti egy, összesen tizenkét 1,5 órás előadást jelent – otthoni munkával kiegészítve –, és azok vehetnek részt rajta, akik már elvégezték a kezdő kurzust. Itt szerzik meg a tanulók azt a tudást és képességet, amelynek segítségével saját tetszőleges programjukat *a lehető legrövidebb idő*

alatt, optimális módon tudják elkészíteni. Összességében tehát ezzel a módszertani megoldással egy év alatt meg lehet tanítani a korábban még soha nem programozó tanulókat a programozás alapjaira úgy, hogy tudásukat azonnal alkalmazni is tudják az iparban.

Az általános iskolák, szakközépiskolák és gimnáziumok oktatási rendjében való alkalmazás annyiban különbözik az itt ismertetett modelltől, hogy ott az otthoni, önállóan elvégzendő feladatokat lecserélik a *tantermi gyakorlatokkal*. Itt, tanári ellenőrzés formájában, azonnali visszajelzést

kapnak a gyerekek a munkájukkal kapcsolatban. A közös, felügyelt foglalkozások lassabb munkát eredményeznek az otthoni feladatmegoldáshoz képest, ennek következménye, hogy a kurzusok hossza fél év helyett egy

tanév hosszúságúra növekszik, vagyis a kezdő és haladó kurzus együtt csak két tanév alatt teljesíthető.

Míndez persze csak akkor lehetséges, ha a tanulók motivációja a képzés és az önképzés során sem hagy alább. Ahhoz, hogy ez megvalósulhasson, a játékfejlesztés önmagában nagyon hatékony eszköz, de a tanár aktív figyelve is szükséges a megfelelő eredményhez. Az órákat kis létszámú, legfeljebb 15 fős csoportok számára ideális megtartani, hogy az oktatás személyesebb, közvetlenebb hangvételű lehessen. A tanulók aktív figyelmére

szükség van, hogy számítógépen keresztül képesek legyenek a tanár magyarázatát követni.

egy év alatt meg lehet tanítani a korábban még soha nem programozó tanulókat a programozás alapjaira

a játékfejlesztés önmagában egy nagyon hatékony eszköz, de a tanár aktív figyelve is szükséges

¹ A www.yoyogames.com oldalon mobil fejlesztéssel kapcsolatos információkat is találni.

Egy kurzus tantárgyi kerete célcsoporttól függően más és más lehet. Egyetemi és főiskolai diákok számára tartott órák könnyen beilleszthetők a felsőoktatásban alkalmazott kreditrendszerbe. Általános iskolások, szakközépiskolások és gimnazisták körében szakkör formájában vezethető be könnyedén a kurzus. Az iskolarendszeren kívül eső oktatáshoz egy szervezeti egység alapítása szükséges, amely képes megadni a kurzusok indításához szükséges személyi és tárgyi feltételeket; eszközöket, helyszínt és egyéb forrásokat.

A kurzusok követelményrendszerét otthoni munkák és projektek formájában érdemes meghatározni, mivel az alkalmazott programozást mint képességet csak a megfelelő mennyiségű *gyakorlás* eredményeképpen lehetséges elérni. A dolgozatok és egyéb azonnali számonkérések csupán az elméleti ismeretek ellenőrzésére és a tudás azonnali reflektálására szolgálnak, ugyanakkor a mai internetes keresések keresztül történő programozás alkalmazásához már jóval kisebb szükség van az így elsajátított tudásra. Különösen azon cégek és szervezetek esetében igaz ez, amelyeknél a programozás nem az elsődleges munkakört jelenti, például mérnököknél vagy kutatóknál.

A fél és egy éves kurzusok tanmenete az alapvető médiakezeléshez kapcsolható ismereteket is tartalmazza. A képek kezelése egy programon belül, a képfeldolgozás alapjai, a hangok kezelése, a zeneszerkesztés alapjai, animációk rajzolása és programozása, illetve a videókészítés alapjai mind a témakör részét képezik.

3. A PRÓBAKURZUSOK ELÉGEDETTSÉGVIZSGÁLATÁNAK BEMUTATÁSA

A kutatási munka során több próbakurzust is indítottam konzulensem, *Dr. Molnár György* támogatásával; ezek célja az optimális oktatási modell minél nagyobb pontossággal történő megközelítése volt. A próbakurzusok után elégedettségvizsgálatot végeztünk, levontuk a megfelelő következtetéseket, és kijelöltük a további fejlesztési irányokat.

több BME-, ELTE- és SOTE-hallgató is jelen volt az órákon, de érdeklődtek az asztalos és a műkörmépítő szakmában dolgozók is

Az első próbakurzus 2013 őszén, egy internetes közösségi oldalon keresztül érte el leendő résztvevőit. A hirdetés elsősorban felnőtteket célt; háromszoros volt a túljelentkezés, köszönhetően a számítástechnika és a programozás minden-

napokban betöltött szerepének. A kurzusra igen sokféle területen dolgozó és tanuló résztvevők gyűltek egybe, aminek következményeképp az egyéni előzetes tudásszintek nagy mértékben különböztek, ahogy az előzetes elvárások is. Több BME-, ELTE- és SOTE-hallgató is jelen volt az órákon, de érdeklődtek az asztalos és a műkörmépítő szakmában dolgozók is. Többek között ennek a változatosságnak köszönhető, hogy a kurzus előadásait módosítani kellett: végül az eredeti kezdő modell anyagának rövidített változatát tartalmazták, összesen három alkalomra redukálva. A három alkalom után az idő rövidsége okán gyakorlatorientált számonkérés formájában ellenőriztük a hallgatók megszerzett tudását. Bár ennek sikere erősen függött a hallgatók által befektetett otthoni munka mennyiségétől, a résztvevők által kitöltött kérdőívek arról ta-

núskodtak, hogy a kurzus során rendkívül motiváltakká váltak oktatástechnológiai és játékefejlesztői vonatkozásban egyaránt.

A következő szemeszterben, 2014 tavaszán a próbakurzust a Budapesti Corvinus Egyetemen, a Kinizsi Kollégiumban hirdettük meg egyetemisták és külsősök számára, ahol a kezdő kurzus modelljének 8 alkalmas változatán vehettek részt az érdeklődők. Itt már házi feladatok formájában történt a számonkérés; ezek kidolgozására két

hónap állt rendelkezésre, majd az utolsó előadáson prezentálni kellett az eredményt a többi hallgató számára. A kurzus itt már nemcsak elégedettségben mutatott rendkívül pozitív eredményeket, hanem szakmai tudás tekintetében is jelentős fejlődésen mentek keresztül a hallgatók. Az egyértelmű sikert igazolja, hogy a hallgatók *több mint negyede* a házi feladatok hatására *programozói pályát választott*. Sok személy látott perspektívát a Game Maker program hosszú távú használatában és további oktatási célú fejlesztésében. A visszajelzésekből az is kiderült, hogy akik végigjárták az órákat, nagyon élvezték azt, hogy egyre mélyebben látnak bele egy programozó mindennapi problémáiba. Kivételesen mindenki megemlítette, hogy az oktatásmódszertant alkalmazni kellene szélesebb körökben is, mivel szerintük is *hatékonyabban lehet megérteni ezzel a metodikával az algoritmikus gondolkozásmódot*.

Az ezt követő két szemeszterben, 2014 őszén és 2015 tavaszán a Budapesti Műszaki és Gazdaságtudományi Egyetemen egy meg-lévő, szabadon választható tárgy tematikájába illesztettük be a programot, emiatt azonban a célközönség jelen-

tősen szűkült. Ez alkalommal hat foglalkozásból álló kurzusokat hirdettünk a kezdő modell alapján. Statisztikai adatokkal még nem lehet egyértelműen alátámasztani a hallgatók elégedettségét és a kurzus sikerességét. Ugyanakkor – személyes viz-

szajelzések alapján – ez a kurzus a korábbiakéhoz hasonló hatású: több mérnökhallgató a leadott órák alapján kezdett el részletesebben foglalkozni a programozással. Többen is kifejtették személyes

véleményüket azzal kapcsolatban, hogy a jelenleg működő programozásoktatás módszertana nem sikeres, mert bár nagy mennyiségű elméleti tudásra tehetnek szert a hallgatók, nem tudják azt rögzíteni gyakorlatokon keresztül. Az egyetemi kötelezettségeket ugyan teljesítik, de azok közel sem olyan kreatívak és motiváló hatásúak, mint a próbakurzusok által nyújtott játékefejlesztés, beleértve annak pedagógiai megjelenítését is.

4. KITEKINTÉS

Bár a két éve zajló felmérés az ipari alkalmazhatóságot és az oktatási módszer hosszú távú hatását nem képes még kimutatni, a visszajelzések és a szorgalmi feladatokba

fektetett nagy energia jelzi a motiváció ugrásszerű növekedését. A hallgatók önszorgalomból, a szabadidejük jelentős részében is szívesen foglalkoznak szoftverfejlesztéssel, amely tudás akár a más területen végzett, számítógépes

a hallgatók több mint negyede a házi feladatok hatására programozói pályát választott

a visszajelzések és a szorgalmi feladatokba fektetett nagy energia jelzi a motiváció ugrásszerű növekedését

szoftverekkel kapcsolatos tevékenységüket is segítheti. Érdemes tehát ezzel a területtel

tovább foglalkozni, és szélesebb körben újabb felméréseket indítani. A kialakított új módszertani megközelítés fejlesztését nagyban segítené, ha a későbbiekben induló kezdő kurzus az elméleti összefoglalóban vázolt javaslatoknak maradéktalanul megfelelné, illetve ha a haladó kurzus felmérése is megvalósulhatna.

Elengedhetetlen a megfelelő minőségű *oktatói képzés* is, hiszen az új módszertan szélesebb körben történő alkalmazásához szükség van a megfelelő szakmai háttérismeretekkel rendelkező pedagógusokra. A tanárok között egységesíteni kell a használatos oktatásmódszertant a tananyag fejlesztése, illetve a kurzusok eltérő változatainak rugalmas meghatározhatóság céljából.

Hosszú távon, amennyiben a kutatás már kellő mennyiségű statisztikai adattal

szolgál, az itt vázolt oktatásmódszertant érdemes összehasonlítani a hagyományos programozásoktatási módszerrel. A cél, hogy a kutatómunka eredményeképpen egyaránt javuljon a hallgatók elégedettségi szintje, valamint a programozás oktatásának minősége és hatékonysága.

Ezen célok megvalósítása érdekében a kifejlesztett oktatásmódszertan az egyetemi oktatási szintér mellett budapesti általános iskolákban és gimnáziumokban is meg fog jelenni tanítási órákon kívüli szakkörök formájában a közeljövőben. A kezdeményezés célja a digitális bennszülöttek informatikai képzésének pótlása, javítása. (Edu&Joy, 2015) Emellett az egyetemi oktatás keretében természetesen folytatódik a kutatás, a módszertan fejlesztése és a statisztikai adatok gyűjtése.

IRODALOM

- Anderson, E. F. és McLoughlin, L. (2007): *Critters in the Classroom: A 3D Computer Game-Like Tool for Teaching Programming to Computer Animation Students*. The National Centre for Computer Animation, Bournemouth University.
- Benedek András (2007): Mobiltanulás és az egész életen át megszerzhető tudás. *Világosság*, 9. sz., 21-28.
- Bleed, R. (2005): *Visual Literacy in Higher Education*. Maricopa Community Colleges, ELI Explorations.
- Brain, M. (2009): *Teaching your kids how to write computer programs*. Letöltés: <http://marshallbrain.com/kids-programming.htm> (2013. 10. 27.)
- Chen, W. K. és Cheng, Y. C. (2007): Teaching Object-Oriented Programming Laboratory With Computer Game Programming. *IEEE Transactions on Education*, 50. 3. sz.
- Edu&Joy (2015): <http://eduandjoy.hu/>. Letöltés: 2015. 08. 01.
- Finley, K. (2013): *World of Codecraft: 3-D Game Teaches Kids 'Big Ideas' of Programming*. Letöltés: <http://www.wired.com/2013/07/programming-game-engagement/> (2013. 10. 27.)
- Haden, P. (2006): The Incredible Rainbow Spitting Chicken: Teaching Traditional Programming Skills Through Games Programming. Eighth Australian Computing Education Conference (ACE2006). *Conferences in Research in Practice in Information Technology*, 52. 81-89.
- Heussner, K. M. (2013): *Robots and rocket launchers: Kuato Studios uses gaming to teach kids the basics of coding*. Letöltés: <http://gigaom.com/2013/03/26/robots-and-rocket-launchers-kuato-studios-uses-gaming-to-teach-kids-the-basics-of-coding/> (2013. 10. 27.)
- Katona Krisztina és Kurdi Zsombor Zsolt (2008. 08. 27): A szoftvertervezés gyakorlati oktatása. *IF2008 konferencia*, Debrecen.

- Kirkland, D. és O'Riordan, F. (2008): *Games as an Engaging Teaching and Learning Technique: Learning or playing?*, Griffith College, Dublin.
- Kurzweil (2013): *A video game that teaches how to program in Java*. Letöltés: <http://www.kurzweilai.net/a-video-game-that-teaches-how-to-program-in-java> (2013. 10. 27.)
- Leutenegger, S. és Edgington, J. (2004): *A Games First Approach to Teaching Introductory Programming. Conference'04*, 1–2. sz.
- Lockhart (2013): *Games That Teach Programming: A Brief Overview*. Letöltés: http://www.gamasutra.com/blogs/RobLockhart/20130905/199667/Games_That_Teach_Programming_A_Brief_Overview.php (2013. 10. 27.)
- Mikulok, K. (2013): *Best Programming Languages to Learn in 2013: The Elite Eight*. Letöltés: <https://www.udemy.com/blog/best-programming-language/> (2013. 10. 27.)
- Minden, ami Közgáz (2010): *Oktatás + módszertan = oktatásmódszertan*, Letöltés: http://kozgaz.blog.hu/2010/06/17/oktatas_modszeratan_oktatasmodszeratan (2013. 10. 27.)
- Miskolczi Ildikó (2010): *Tanulás vagy E-tanulás? – Oktatásmódszertan a XXI. században, Szolnoki Tudományos Közlemények*, 14. sz. Letöltés: http://tudomany.szolnok-mtesz.hu/kulonszamok/2010/cikkek/Miskolczi_Ildiko.pdf (2015. 06. 25.)
- Molnár György (2011): *Új módszerek a pedagógiai gyakorlatban – az IKT alapú megoldások tükrében. Szakképzési Szemle*, 27. 3. sz. 170-177.
- Molnár György (2012): *A technológia és hálózatalapú tanulási formák és attitűdök az információs társadalomban, különös tekintettel a felsőoktatás bázisára. Információs Társadalom*, 3. sz. 61-76.
- Molnár György és Benedek András (2013): *ICT Related Tasks and Challenges In The New Model of Technical Teacher Training*. In: John Terzakis, Constantin Paleologu, Tibor Gyires (szerk.): *InfoWare 2013: ICCGI 2013: The Eighth International Multi-Conference on Computing in the Global Information Technology*, Nizza. 40-44.
- Molnár György (2014): *Pedagógiai megújulás tapasztalatai a szakmai tanárképzésben – új IKT alapú eszközök és koncepciók a tanárképzésben*. In: Ollé János (szerk.): *VI. Oktatás-Informatikai Konferencia Tanulmánykötet.*, ELTE PPK Neveléstudományi Intézet, Budapest. 434-452.
- Nyíri Kristóf (2003): *From Texts to Pictures: The New Unity of Science*. In: Nyíri, K. (szerk.): *Mobile Learning: Essays on Philosophy, Psychology and Education*. Passagen Verlag, Bécs. 45-67.
- Overmars, M. (2004): *Teaching Computer Science through Game Design. Computer*, 4. sz. 81-83.
- Pap-Szigeti Róbert és Pásztor Attila (2008): *A LEGO programozható robotjaival segített programozásoktatás bevéltésvizsgálata*. In: Pethő Attila és Herdon Miklós (szerk.): *Informatika a felsőoktatásban 2008*. Debreceni Egyetem, Informatikai Kar, Debrecen. Letöltés: <http://www.agr.unideb.hu/if2008/dok/absztrakt.pdf> (2015. 06. 25.)
- Pinola, M. (2013): *How and Why to Teach Your Kids to Code*. Letöltés: <http://liferhacker.com/how-and-why-to-teach-your-kids-to-code-510588878> (2013. 10. 27.)
- Primerlabs (2013): Letöltés: <http://primerlabs.com/> (2013. 10. 27.)
- Rajaravivarma, R. (2005): *A games-based approach for teaching the introductory programming course. Newsletter ACM SIGCSE Bulletin*, 37. 4. sz. 98-102.
- Romano, B. (2013): *Dan Shapiro Builds a Board Game to Teach 3-Year-Olds Programming*. Letöltés: <http://www.xconomy.com/seattle/2013/09/03/dan-shapiro-builds-board-game-teach-3-year-olds-programming/> (2013.10. 27.)
- Siemens, G. (2005): *Connectivism: A Learning Theory for the Digital Age*. Letöltés: http://www.itdl.org/Journal/Jan_05/article01.htm (2013. 10. 27.)
- Siemens, G. (2006): *Knowing Knowledge*. Letöltés: http://www.elearnspace.org/KnowingKnowledge_LowRes.pdf (2015. 06. 25.)
- Tóth György Ferenc és Johanyák Zsolt Csaba (2007. 04. 12.): *A szoftverfejlesztés oktatásának reformja a versenyszféra visszajelzései alapján. V. Pedagógiai Értékelési Konferencia*, Szeged.
- Visual Education (2013): Letöltés: <http://centerforvisualeducation.com/about-vised/> (2013. 10. 27.)
- Yoyo games (2015): Letöltés: <http://www.yoyogames.com/> (2015. 09. 05.)