

Szimulációs modellekkel támogatott programozás tanítása az alapiskolában

Stoffová Veronika¹, Czakoóvá Krisztina²

¹NikaStoffova@seznam.cz

Nagyszombati Egyetem, Tanárképző Kar

²czakoovak@uj.s.sk

Selye János Egyetem, Gazdaságtudományi Kar

Absztrakt. A szimulációs modellek alkalmazása jól kidolgozott és gyakran használt kutatási eljárás az egzakt, főleg műszaki tudományok területén. Új rendszerek megismerésére, beazonosítására, működési elveiknek feltárására és így a modellezett (reális vagy fiktív) objektumokról új tudásszerzésre ad lehetőséget. A matematikai modellen alapuló szimulációs kísérletek nélkülözhetetlenek olyan dinamikus folyamatok esetében, amikor az elemzett esemény veszélyes, idő- vagy anyagi befektetésre nézve igényes, esetleg túl gyorsan játszódik le vagy szabad szemmel nem megfigyelhető stb. Az utóbbi időben a szimulációk jelentős szerepet kaptak a mély tanulásban, az úgynevezett „deep learning”-ben. A jól megtervezett algoritmussal irányított vizualizált szimulációs modellel végzett kísérletek által a tanuló (diák, hallgató) számára fontos, a tanulás témáját érintő ismeretekre tehet szert, amit hasznosíthat a témával kapcsolatos feladatok megoldásában egyaránt. A saját tapasztalatok, a megfigyelés és meggyőződés alapján szerzett ismeretek mélyebbek, tartósabbak és rendszerezettebbek. Így a tanuló könnyebben fedezi fel az összefüggéseket, aktívan tudja használni és egyúttal felépíteni saját tudásrendszerét. A tanulmány két szimulációs játékot is bemutat, mely a programozás játékos tanítását hivatott támogatni az alapiskolán.

Kulcsszavak: programozás, programozás tanítása, programozás tanulása, mély tanulás

1. Bevezető a modellezés és szimulációba

A modellezés és szimuláció erős oktástechnológiai eszközzé fejlődött. Számos tantárgyban alkalmazható, és nem utolsósorban a mély tanuláshoz vezet. Főleg olyan témakör tudásanyagának elsajátításban érvényesíthető, ahol nélkülözhetetlen az összefüggések megértése és azok egzakt kifejezése. A viszonylag egyszerű matematikai modell alapján számos dinamikus jelenség és folyamat is bemutatható, és így könnyebben elérhető a kítűzött didaktikai célok.

Olyan jelenségekre és folyamatokra gondolunk, amelyek már ismertek, matematikailag pontosan leírhatók és az oktatás tárgyát képezik. A paraméterek által vezérelt animációkkal be lehet szemléletesen mutatni a szimulációs kísérletek eredményeit. A jól megtervezett és pontosan végrehajtott szimulációs kísérletek lehetővé teszik a modell paramétermódosításai által a modellezett objektum viselkedésére gyakorolt hatás tanulmányozását. Alternatív megoldásként a szimulációs kísérleteket fel lehet használni a modellezési folyamatok tulajdonságainak feltárására (felfedezésére) és tesztelésére, amelyet a felhasználó/kutató/tanuló saját megfigyelései alapján végez. A dinamikus szemléltetés, a vizuális ábrázolás és a grafikus kifejezés elősegítik a kísérlet eredményeinek helyes megértését és azok helyes értelmezését.

A szerzők a szimulációt főképp, mint tanítási és tanulási módszert kívánják bemutatni. A szimulációs modellek a kísérletek eredményei alapján a modellezett objektummal kapcsolatos új ismeretek megszerzésére szolgálnak. A műszaki, természettudományi, gazdaságtudományi és más egzakt tudományok területén a megszerzett tudás relevanciáját a tanulmányozott tárgy esetébe a rendszer

pontos matematikai modellje biztosítja. A módszer korlátlan lehetőségeket ad a modellezett rendszer – a tanulás tárgyának megismerésére [1], [2], [3], [5].

A jól megvalósított matematikai modell helyettesítheti a valós objektumot, és lehetővé teszi annak tulajdonságainak tanulmányozását szimulációs kísérletek segítségével, hogy a megfigyelő/tanuló új ismereteket szerezzen a modellezett objektumról.

1.1. Mi is valójában a modellezés és a szimuláció?

A szimulációs modell segítségével információkat kaphatunk arról, hogyan fog viselkedni valami anélkül, hogy valóban tesztelésre kerülne a reális életben. Például abban az esetben lehet segítségünkre, ha egy versenyautót akarunk tervezni, és nem tudjuk pontosan, milyen funkcionális részeket építsünk bele. Az autó számítógépes szimulációjával képesek vagyunk felbecsülni a különböző alkatrészek, formái megoldások és egyéb paraméterek eredményre gyakorolt hatását. Hasznos betekintést nyerhetünk a különböző döntésekbe, amelyeket az autógyártásnál hozhatunk anélkül, hogy az autó prototípusát elkészítenénk.

Általánosabban fogalmazva, a szimuláció modelleket alkalmaz, beleértve emulátorokat, prototípusokat és szimulátorokat a (tervezett) objektum adatai és paraméterei kidolgozása érdekében, vezetői vagy műszaki döntések meghozatalához.

A „modellezés” és a „szimuláció” kifejezéseket gyakran felcserélhetően szinonimaként használják – ez azonban nem mindig helyes. A szimuláció a műszaki gyakorlatban a modellel végzett szimulációs kísérleteket jelenti [8].

1.2. A modellezés és szimuláció, mint tudományág

A modellezés és a szimuláció (M&Sz) fontos szerepet játszik a tudományos kutatásban. A valós rendszerek reprezentálása – akár fizikai reprodukciók révén (kisebb méretben), akár matematikai modellek segítségével, amelyek lehetővé teszik a rendszer dinamikájának szimuláción keresztüli ábrázolását – lehetővé teszi a rendszer viselkedésének olyan szintű feltárását, amely gyakran nem lehetséges, vagy túl kockázatos lenne a valóságban.

A M&Sz használata a mérnöki kutató, fejlesztő és irányító munkák során bevált és elismert. A szimulációs technológiák az összes alkalmazási szakterületeken felhasználhatók, beleértve a mérnöki menedzsment szakterületét is, hiszen a szimulációs modellek helyes alkalmazásával hozzájárulhatnak a költségek csökkentéséhez, a termékek és rendszerek minőségének javításához. Annak biztosítása érdekében, hogy a szimuláció eredményei alkalmazhatók legyenek a reális életben, egy menedzsernek meg kell értenie eme feltörekvő tudományág feltételeit, koncepcióit és megvalósítási korlátait egyaránt [8].

1.3. A modellezés és szimuláció előnyei

A szimulációs alkalmazások iránt az érdeklődés folyamatosan növekszik. Ennek okai a következők:

1. A szimulációk használata általában olcsóbb és biztonságosabb, mint a termék prototípusán végzett kísérletek.
2. A szimulációk lehetővé teszik egy virtuális környezet létrehozását, amely megengedi a szimulált rendszerek megvizsgálását, részletes elemzését, még a tervezési szakaszban. Majd vegyes virtuális rendszerek segítségével, beépítve az első prototípusú komponenseket, a tervezett rendszert tesztelni és optimalizálni lehet virtuális környezetében. És ezt mind még az az első funkcionális rész kiépítése előtt.
3. A szimulációk gyakran gyorsabban hajthatók végre, mint azok a valós időben megtörténnek. Ez lehetővé teszi a különféle alternatívák hatékony elemzését, különösen akkor, ha a szimuláció inicializálásához szükséges adatok könnyen beszerezhetők az operatív adatokból. A szimulációk

- gyakran valóságghűbbek is lehetnek, mint a hagyományos kísérletek, mivel lehetővé teszik a végeredmék operatív alkalmazási területén található környezeti paraméterek szabad konfigurálását.
4. Az M&Sz elmélete és gyakorlata szorosan összefügg és kölcsönösen támogatja egymást, és így meghatározzák a M&Sz tudományos alapjait.
 5. Az M&Sz elmélete folyamatosan fejlődik és keresi az alkalmazható megoldási mintákat. A hangsúlyt az általános módszerek jelentik, melyek különféle problémakörökben alkalmazhatók.
 6. Az M&Sz alkalmazások úgy oldják meg a valós világ problémáit, hogy megoldási lehetőségekre és stratégiákra összpontosítanak. Gyakran a megoldás csak egy módszer alkalmazásából származik. Sok esetben hangsúlyosan problémakör-specifikus, és a problémakör-szakértelemből vezetik le, nem pedig az általános elméletből vagy módszerből.
 7. A modellek különféle egységekből (finomabb felbontású részletes modellekből) összekapcsolhatók egy adott cél elérése érdekében, ezért modellező megoldásoknak is nevezhetők [8].

1.4. A modellezés és szimuláció folyamata

A modellezés folyamata iteratív jellegű. Magába foglalja a következő lépéseket, amelyek szakaszonként vagy egyenként is megismételhetők (javíthatók):

- a modellezés tárgyának meghatározása;
- a modellezés céljainak meghatározása;
- az objektum, mint rendszer azonosítása (a rendszer bevezetése a kiválasztott objektumon);
- az alrendszerek és a részletek szintjének meghatározása;
- a modell matematikai leírásának meghatározása;
- matematikai modellek implementálása a számítógépen;
- szimulációs kísérletek megvalósítása;
- a szimulációs eredmények értelmezése [8].

2. Szimulációs játékok algoritmizálás és programozás tanítására

Ebben a fejezetben két játékot, illetve a játékos programozás tanítására szolgáló alkalmazást mutatunk be [7], [9], [10], [11], [12], melyek az alapiskolai programozás oktatására és a problémák megoldásának algoritmizálására fókuszál. A programozás tanításában és tanulásában fontos a feladat megoldását standard algoritmusok alapján kifejezni. Ennek a begyakorlását szolgálhatják a következő alfejezetekben bemutatott interaktív szimulációk.

2.1. Stratégiafejlesztés alapú játék az alapvető vezérlési szerkezetek megértésére

Az itt bemutatott játék a szekvencia (utasítások egymás után történő végrehajtása), az elágazás (a feltételtől függően a végrehajtható ágak száma akár több is lehet) és a ciklus (adott számú ismétléssel) programrészek (programnyelvi konstrukciók) megértésére és aktív használatára irányul. A számítógépes játék három nehézségi szintre épül, amelyek a játék folyamán egymást követik a kapott feladat (vizuálisan megjelenített szimulációs modell) sikeres megoldását követően. A játékfelület engedélyezi a kínált utasítások (lépés előre, hátra, forgás 90 fokkal jobbra vagy balra) szekvenciába való foglalását, majd az összeállított program megvalósítását és vizuális kiértékelését [7].

A kalózlány vagy kalózfű vezérlésére szolgáló ikonok, amelyek a játék képernyőjének bal alsó sarkában találhatóak, a programozható játékok irányítására szolgáló nyomógombokra emlékeztetnek. Így a diákok kamatoztathatják a jól ismert programozható méhecske (Bee-bot) vagy programozható autó (Pro-bot) használata során szerzett tapasztalataikat és tovább fejleszthetik képességeiket algo-

ritmus (feladatmegoldás menet) írásból. A következő 3 képernyőképből világos a játék menete, valamint interaktív és intuitív kezelése. A program a jobb felső sarokban kerül ábrázolásra. Itt jelenik meg a program helyességének vizuális értékelése is.



1. ábra: 1. pálya – szekvencia kialakítása elemi utasítások által



2. ábra: 2. pálya – számláló ciklusvezérlés és elágazás megvalósítása



3. ábra: 3. pálya – nehezítéssel, akadályok leküzdésével

Az előbbi ábrák a játék menetét és kezelését illusztrálják. Az első ábra egy egyszerű útvonal bejárását mutatja. A szekvencia begyakorlására szolgál. A második ábrán látható, hogy a játékpálya bejárását számláló ciklus alkalmazása teszi gyorsabbá. A pálya végén feltételes elágazással is szembesülhetünk, melynek feltétele a több drágakő megszerzése. Az út bejárását (programot) a jobb felső sarokban elhelyezett ablak mutatja a kiválasztott lépések grafikus megjelenítésével. Az ábrán az is látható, hogy az egyes vezérlő struktúrák (parancs szekvencia – szürke, ciklussal kiválasztott utasítás-sorozat – kék) más-más színnel vannak megjelenítve. A harmadik ábrán a bejárando útvonal nehezítéssel, azaz akadályokkal (harcos kalózzokkal) került kibővítésre. Ezeket jobb elkerülni, hogy ne kelljen a rossz kalózzal szembesülni, néhányat kivéve, akikkel való megküzdés jutalma a félfúton megszerzhető drágakövekben lakozik [7].

A játék (applikáció) a *Visual Studio 2015* fejlesztőkörnyezetben készült, C++ programnyelv segítségével. A játék grafikai elemeinek szerkesztéséhez a Paint.net rajzprogram szolgált.

2.2. A ciklusok játékos tanítása

A következő alfejezetben bemutatásra kerülő interaktív szimulációk célja a különböző ciklus típusok értelmezése, azok megértése, különös tekintettel az algoritmusok és programozás témakör oktatása során [7], [12], [13].

Az itt leírt interaktív szimulációk egységes grafikus szerkezettel rendelkeznek (szimulációs feladat megnevezése, rövid leírása, a megoldás algoritmusának folyamatábrával és programkóddal való kifejezése és az eredmény vizualizálása). A feladat megnevezése és rövid leírása az ábrákról hiányzik, mivel e szimulációk eredetileg szlovák nyelven készültek. Így azok feltüntetésétől eltekintettünk (a szlovák nyelvet nem értő olvasó számára ezen adatokat elhanyagolhatónak ítéltük). Remélhetőleg az ábrákból így is világosan látszik az interaktív szimulációk kezelése, az információk formájának kifejezése és a rész-/eredmények vizualizálása, ami a szimulációk lényeges küldetése.

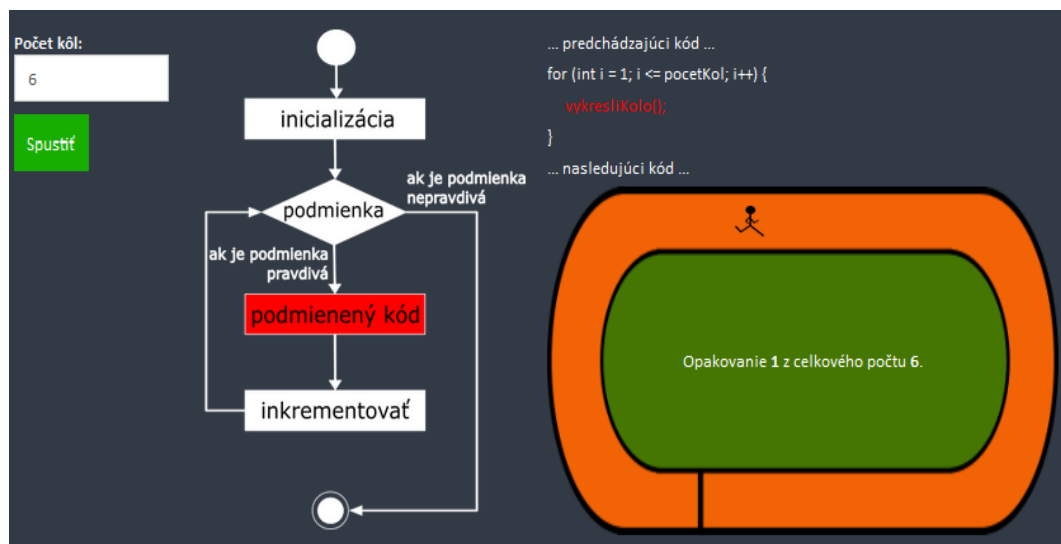
Minden animációhoz felhasználói kézikönyv is készült, amely módszertani anyagként szolgál tanítók/tanárok számára az adott szimuláció megfelelő alkalmazására az oktatásban.

2.2.1. Megadott számú ismétlés – for típusú ciklus

A **for ciklus** működési elveit egy életről való példa mutatja. A futó a felhasználó (tanuló) által megadott számú kört fut le egy futballpálya körül. A negyedik ábrából világos a szimuláció paramétereinek beállítása és a feladat elvégzésének nyomon követése [4].

A szimulációs modell paramétere egy - a megtett körök számát képviselő - pozitív egész szám, amely alapján a futó lefutja az adott számú kört. Ebben a szimulációban a végkimenet három formában került megjelenítésre: animációval, folyamatábrával és program formájában.

Az algoritmus egyes lépéseit a szimuláció pillanatnyi állapotát követve figyelhetjük meg, a kód kiemelt része, a folyamatábra, illetve a futó mozgása által. A szimuláció vezérlésére az Indítás (**Spustíť**) gomb szolgál [4].

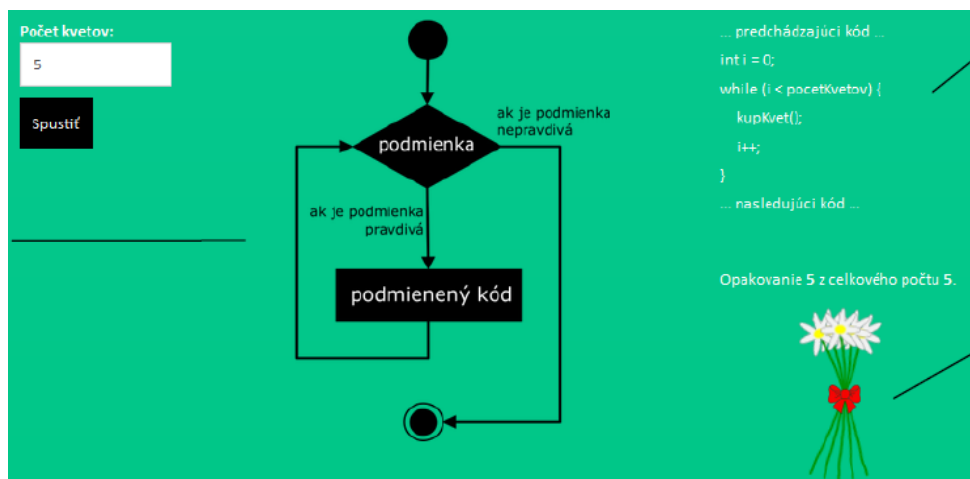


4. ábra: Adott számú ismétlés – for típusú (számláló) ciklus szemléltetése

2.2.2. Ismétlés előfeltétellel – előltesztelő ciklus (while típusú ciklus)

A **while típusú ciklus** működési elvét a szimuláció egy csokor elkészítése példáján mutatja be. A felhasználó megadhatja, hány virágot kíván csokorba kötni. Az ötödik ábrából világos a szimuláció paramétereinek beállítása és a feladat végrehajtásának nyomon követése.

A szimuláció létrehozása és felépítése hasonló az előző szimulációhoz. Grafikus (animált) objektumként szerepelnek: a virág, a szalag, és a folyamatábra [4]. Azt a feltételt kell megadni, amellyel a csokor a virág kirajzolása által van (ismétléssel) irányítva (5. Ábra). A virágok csokorba fűzését virág alakú kép fokozatos kirajzolása biztosítja, különböző elforgatással. Az első virágot függőlegesen felfelé rajzoljuk, és minden ezt követő virágot 10° -kal elforgatjuk (felváltva minkét irányba), figyelembe véve a virágok egyenletes eloszlását a csokron belül. A ciklus szükséges ismétlésének elvégzése után befejezzük a csokor kirajzolását és a csokrot egy szalaggal megkötjük.



5. ábra: Az előtesztelő ciklus szemléltetése

2.2.3. Ismétlés utófeltétellel – hátulatesztelő ciklus (repeat - until típusú ciklus)

A **repeat - until típusú ciklus** működési elveit a szimuláció folyadékelegítés példáján mutatja be. Többféle előre definiált folyadékából lehet választani (pl. víz, olaj stb.). A kiválasztott folyadék alapján a program automatikusan hozzárendeli a hozzá tartozó szükséges paramétereket (sűrűség, fajhő). A felhasználó definiálja még a többi belépő adatot, mint: a kezdőhőmérsékletet, a vég hőmérsékletet, a melegítő spirál teljesítményét és a folyadék térfogatát [4].

A folyadékelegítés szimulációjával ábrázoljuk a ciklus menetének folyamatát, amely a ciklus lefutása után teszteli a befejezést/leállítást irányító feltétel teljesülését. A ciklust ugyanúgy határozzuk meg, mint az előtesztelő ciklusnál. A különbség azonban lényeges. A hátulatesztelő ciklus esetében a ciklustest rész legalább egyszer lefut, mivel az első tesztelés az első végrehajtás után valósul meg. Amint a meghatározott ciklusfeltétel teljesül, a program leáll. Az algoritmus alapján a szimuláció addig melegíti a folyadékot, amíg az el nem éri a felhasználó által meghatározott vég hőmérsékletet (6. Ábra). A példa fizikai ismeretekre épül, így a programozás mellett a fizikából szerzett jártasságokat és készségeket is próbálja fejleszteni.

A 2.2 alfejezet alatt bemutatott interaktív szimulációk a HTML, CSS, JavaScript használatával készültek, így alkalmasak web böngészőbe való betöltésre és futtatásra [4], [14], [15], [16].

3. Befejezés

A cikkben rámutattunk az interaktív animált szimulációk jelentőségére és hozadékára a mély tanulásban. A pedagógiai megfigyeléseink, több éves tapasztalataink a programozás tanításában és az elvégzett pedagógiai kísérletek eredményei is alátámasztják azt, hogy az animációkkal kísért szimulációknak fontos szerepük van a programozás és algoritmizálás oktatásában. Algoritmussal irányított animációk segítségével a tanulók könnyebben, rövidebb idő alatt, sokszor izgalmas, játékos formában érthetik meg a szemléltetett fogalmakat, folyamatokat és a köztük lévő különbségeket egyaránt [15].

Fontos azonban megemlíteni, hogy egy új animáció megtervezésénél, ill. fejlesztésénél figyelembe kell venni a multimédiával támogatott tananyag didaktikai elveinek betartását és a megfelelő grafikus reprezentáció megválasztását. Fontos, hogy megfelelő interaktivitást iktassunk az animációkba.

A gondosan megtervezett, implementált és tesztelt interaktív szimulációk hasznos oktatási segédeszközként szolgálhatnak úgy a tanítók, mint a diákok számára.

... predchádzajúci kód ...
 int i = 0;
 do {
 varvodu();
 i++;
 } while(i < koncnaTeplota)
 ... nasledujúci kód ...

Opakovanie 7 z celkového počtu 7.

$$T_1 = \frac{t \cdot P}{V \cdot \rho \cdot c} - T \quad [^{\circ}\text{C}]$$

Opakovanie číslo	Čas zohrievania [t]	Aktuálna teplota [T ₁]
1.	30 s	4,43 °C
2.	60 s	4,86 °C
3.	90 s	5,29 °C
4.	120 s	5,73 °C
5.	150 s	6,16 °C
6.	180 s	6,59 °C
7.	208,58 s	7 °C

6. ábra: A hátultesztelő ciklus szemléltetése

Köszönetnyilvánítás

A cikk a KEGA 012TTU-4/2018 Interactive animation and simulation models in education (Interaktív animációs és szimulációs modellek az oktatásban) nemzeti projekt támogatásával készült.

This work has been supported by the Grant Agency of the Slovak republic KEGA under the Grant No. 012TTU-4/2018 Interactive animation and simulation models in education.

Irodalom

1. Czakóová, K. 2016. *Creation small educational software in the micro-world of small languages*. In: Teaching Mathematics and Computer Science. 14th volume, issue one, 2016/1, p. 117. Debrecen: University of Debrecen, 2016. ISSN 1589-7389
2. Czakóová, K.. 2019. *Interaktív modellek és szimulációk az oktatásban*. In. XXXII. Didmattech 2019. Trnava: Trnavská univerzita v Trnave, 2019
3. Czakóová, K.. 2019. Mathematical model based interactive simulations in education. ICERI 2019, 11-13. november 2019, Seville
4. Lapšanská, Š. 2016. *Visualizácie dinter active simulation models for teaching*, Bachelor thesis, Trnava University in Trnava, Faculty of Education, Department of ComputerScience. Supervisor of the bachelor thesis: prof. Ing. Veronika Stoffová, CSc. Trnava: Faculty of Education TU, 2016, 56 p
5. Pokorný, M. 2019. *Blended learning can improve the results of students in combinatorics and data processing*. In: ISET 2019. Los Alamitos: IEEE Computer Society, 2019, s. 207-210. ISBN 978-1-7281-3387-4

6. Pokorný, M. – Oubrechtová, S. 2016. *Blended Learning Can Improve the Results of Students in Mathematics*. In: Information, Communication and Education Application, Advances in Education Research, Volume 94, 2016, s. 57-62. ISBN 978-1-61275-505-2, ISSN 2160-1070
7. Szalai, D. 2017. *Algoritmikus gondolkodás és problémamegoldás fejlesztése interaktív játékok segítségével az alapsiskolán*. Diplomamunka, Komárno: Selye János Egyetem, Tanárképző Kar, diplomamunka vezetője: PaedDr. Czakoóvá Krisztina, PhD. 2017, 62 p.
8. Stoffová, V. 2004. *Počítač – univerzálny didaktický prostriedok*. Nitra: Univerzita Konštantína Filozofa, 2004. 172 s. ISBN 80-8050-765-1.
9. Stoffová, V. 2016. *The Importance of Didactic Computer Games in the Acquisition of New Knowledge*. In: The European Proceedings of Social & Behavioural Sciences EpSBS. 2016a, pp. 676-688. eISSN: 2357-1330. <http://dx.doi.org/10.15405/epsbs.2016.11.70>
10. Stoffová, V. 2016. *Počítačové hry a ich klasifikácia*. In: Trendy ve vzdělávání. 2016b, roč. 9, č. 1, s. 243-252.
11. Stoffová, V. - Horváth, R. 2017. *Didactic computer games in teaching and learning process Else Bucurest*, The 13th International Scientific Conference, eLearning and Software for Education, Bucharest, April pp. 27-28, 2017, 10.12753/2066-026X-17-000 DOI: 10.12753/2066-026X-17-000
12. Végh, L. 2016. *Interaktívne animácie vo vyučovaní algoritmov (Interactive animations in teaching and learning programming)*. Edukacja – Technika – Informatyka (Education – Technology – Computer Science), 2016. 15(1): pp. 207-211.
13. Végh, L. 2006. *Vizuálna animácia algoritmov vo vyučovaní programovania*. Informatika v škole a v praxi. Ružomberok: Katolícka univerzita, 2006, s. 65-69. ISBN 80-8084-112-8
14. Végh, L. 2016. *Javascript library for developing interactive micro-level animations for teaching and learning algorithms on one-dimensional arrays*. Acta Didactica Napocensia, 9(2), 23–32.
15. Végh, L. 2017. *A programozás tanulásának és tanításának támogatása elektronikus tananyagba beépíthető interaktív animációs modellekkel*. (PhD theses). Eötvös Loránd Tudományegyetem Informatika Kar, 2017
16. Végh, L. - Stoffová, V. 2017. *Algorithm animations for teaching and learning the main ideas of basic sortings*. In: Informatics in Education, Lithuania : Vilnius University. Vol. 16, No. 1, 2017, p. 121-140. ISSN: 1648-5831 (printed), 2335-8971 (online). DOI: 10.15388/infedu.2017.07