

Informatikai gondolkodás fejlesztésének dimenziói

Pluhár Zsuzsa

pluharzs@inf.elte.hu

Eötvös Loránd Tudományegyetem, Informatikai Kar

Absztrakt. Az informatikai gondolkodás fejlesztése nem csupán számítógéppel, programozással lehetséges. Több nemzetközi és hazai törekvés, irányzat létezik, melyek a számítógép nélküli megoldásokat kutatják és támogatják. Az egyik ilyen a nemzetközi bebras kezdeményezés, mely immár tíz éve van jelen Magyarországon is. Ezidő alatt nem csak az alapokat jelentő verseny megszervezésére került sor minden évben – 2019-ben már több, mint 27ezer diák részvételével – , de a háttérben az algoritmikus gondolkodás mérésére és fejlesztésére használt tesztek és kurzusok kidolgozására, kiegészítő aktivitások kialakítására is. Jelen tanulmány az e-hód kezdeményezés tíz éve alatt folytatott kutatásokat, illetve ezek egymásra épülését foglalja össze.

Kulcsszavak: informatikai gondolkodás, algoritmikus gondolkodás, számítógép nélkül (cs unplugged), bebras, hód

1. Informatikai gondolkodás vizsgálatának dimenziói

Az informatikai gondolkodás (computational thinking) fogalma Wing [1] megfogalmazása óta sokat alakult, formalizálódott.

A fejlesztéséről, méréséről folyó kutatások alapja minden esetben annak megfogalmazása, mely dimenzióit emeljük ki, illetve az egyes dimenzióit hogyan tudjuk szétválasztani. A leginkább elterjedt elmélet [2, 3] három dimenzióra osztja az informatikai gondolkodást: (1) az elméleti fogalmak – program struktúrák és használatuk – ismerete (CT concepts), (2) a gyakorlati ismeretek (practices) mint a problémamegoldás, a probléma megfogalmazása, a dekompozíció, az absztrakció, a tesztelés és hibakeresés folyamatai, és a (3) perspektívák (perspectives), a körülöttünk lévő világ megismerése és megértése, kritikus gondolkodás.

A felhasznált eszközök tekintetében három nagyobb vizsgálati csoportot különíthetünk el:

Az első esetében a programozás, a programozási folyamatok kerülnek hangsúlyozásra – a programkészítés folyamatán, a programozási lépéseken, használt program struktúrák megismerésén, a tervezéstől a tesztelésig végzett tevékenységeken keresztül kerül sor az absztrakció, a dekompozíció, az algoritmikus gondolkodás fejlesztésére. Ebben a csoportban a harmadik (3) dimenzió általában kisebb hangsúlyt kap, van, hogy teljesen kimarad. [4-6]

A második csoport projektfeladatok, projektmunkák során beintegrálja az informatikai gondolkodás elemeit. Ebben az esetben a harmadik dimenzió (3) kap nagyobb szerepet és a programozási elemek, elméleti fogalmak (1) sokszor nem is jelennek meg, vagy kisebb hangsúllyal szerepelnek. A kutatások többsége a STEAM (science, technology, engineering, art and math, azaz természettudományok, technológia, mérnöktudományok, művészet és matematika) területeken fogalmazza meg a projekteket, aktivitásokat. [7-9]

A harmadik csoportba sorolhatjuk azokat a kezdeményezéseket, melyek számítógép nélküli aktivitásokkal valósítják meg az informatikai gondolkodás fejlesztését, vizsgálatát. Ilyen kezdeményezések pl. a legkisebbeknek szóló cs unplugged [10], az idősebbeket megszólító cs4fun [11] vagy a legszelebb életkori és dimenziókat összefoglaló bebras [12, 13].

2. e-hód

A bebras kezdeményezés céljait [12, 13] kiterjesztve, illetve testreszabva fogalmazzuk meg a magyarországi megvalósulás, az e-hód célkitűzéseit [15, 16].

A feladatok, aktivitások témájában és megfogalmazásában fontos, hogy:

- felkeltse az érdeklődést az informatika iránt;
- feloldja az informatikával kapcsolatos félelmeket, negatív érzéseket;
- megmutassa az informatika területének sokszínűségét, felhasználási lehetőségeit és területeit.

Míndezt úgy, hogy informatikai előképzettséget nem igényel.

Célsoporként nem csupán a diákokat nevezhetjük meg, de ugyanilyen fontosak számunkra a közoktatásban résztvevő és a hamarosan bekerülő tanárok is. Az ő számukra célunk támogatást adni az egyes informatikai témakörök motiváló probléma-felvetéseihez, példa alapú megoldásaihoz, valamint az informatika sokszínűségének és integrálásának lehetőségeihez.

A kezdeményezés keretein belül létrehozott anyagok többségére a CC BY-NC-SA 4.0 licenz vonatkozik, így a közoktatásban résztvevők könnyen felhasználhatják azokat.

2.1. Verseny

A kezdeményezés alapjait egy verseny jelenti, melyben 18 rövid, gyorsan (kb. 3 perc alatt) megoldható, informatikai előképzettséget nem igénylő feladatot kell megoldaniuk a résztvevőknek.

A feladatok három nehézségi szinten (nehéz, közepes és könnyű) megfogalmazott, érdekes problémákat mutatnak be. Nem tesztek, inkább szórakoztató gondolkodtató feladványok, melyekkel a résztvevők új ismeretekre tehetnek szert, illetve meglévő ismereteiket mélyíthetik el.

A feladatok előkészítését, pontosítását a nemzetközi csapat egy műhelykonferencia keretein belül végzi, kiválogatva és módosítva a résztvevő országokból beküldött kérdéseket. Ezután az egyes országok honosítják a kérdéseket, és a náluk megrendezett versenyhez testre szabják azokat.

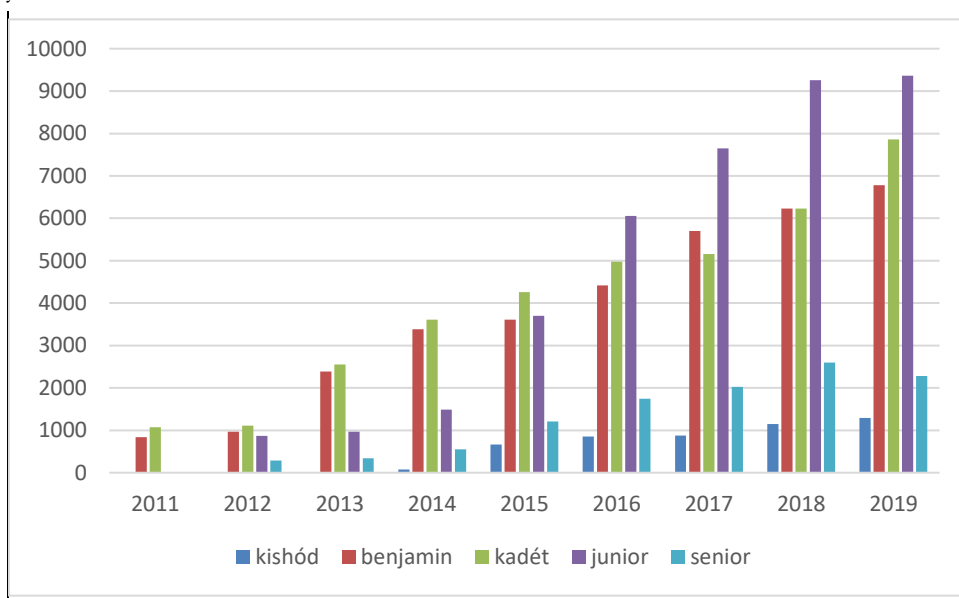
2010 óta Magyarország is – az ELTE IK vezetésével és tanárszakos hallgatók bevonásával – részt vállal ebben az előkészítő folyamatban úgy, hogy 2011 óta megszervezzük a versenyt is. 2014 óta pedig mint National Bebras Organiser teljes jogú tagjai lettünk az International Bebras Committee-nek.

Az első alkalommal 2 életkori kategóriában (benjamin, kadét) 1907 diák versenyzett 24 iskolából. 2019-re 202 iskola több, mint 27ezer diákja (27 570) öt életkori kategóriában mérte meg magát.

Korcsoport kategória	Osztály
kishód	4.
benjamin	5–6.
kadét (első évben meteor)	7–8.
junior	9–10.
senior	11–12. (13.)

1. táblázat: Életkori kategóriák a hód versenyeken.

Az egyes életkori kategóriákban (1. táblázat) résztvevők számának alakulásából (1. ábra) leolvasható az is, hogy a fiatalabbak átlépnek az idősebb korosztályos csoportokba, viszik magukkal a versenyt.



1. ábra: Résztvevőszámok alakulása életkori kategóriánként a magyarországi hód versenyeken.

A verseny országos lefedettsége egyre erőteljesebb.

A versenyek eredményességéről, célkitűzéseink megvalósulásáról több fronton igyekszünk megbizonyosodni. Folyamatosan fogadjuk a résztvevők visszajelzéseit. Nem csak kötetlen formában, de minden verseny után kérdőívet is készítünk, melyet a koordinálást segítő tanárok önkéntesen töltenek ki és küldik el számunkra nem csak a saját, de diákjaik észrevételeit, tapasztalatait is.

Időről időre kérdőíves kitöltésre kérjük fel a diákokat is, melynek segítségével igyekszünk vizsgálni, hogy a sikeres részvételüket milyen dimenziók, paraméterek segítik, illetve a részvételük a versenyen milyen egyéb területen fejleszti őket, ad pluszt számukra.

Minden évben, a teljesítmények tükrében megvizsgáljuk a kérdések erősségét, besorolásuk helyességét is. Vizsgáljuk azokat a helytelen válaszlehetőségeket, melyeket kiemelkedően sok diák választ ki egy-egy feladatnál.

Összességében – korábban is publikált eredményeinket tekintve [15, 16] – elmondhatjuk, hogy a kérdések nehézségi szintje megfelelő, illetve általánosan jó nehézségi szint besorolást kapnak a verseny során.

A nemek szerinti bontásban sem az egész minta vonatkozásában, sem a korcsoportonkénti bontásban nem látható szignifikáns különbség a fiúk és lányok teljesítményében egyik évben sem, valamint a versenyen való sikeres szereplést leginkább befolyásoló tényező a korábbi részvételek száma volt.

A résztvevőszámok alakulásából következtetve, illetve a kötetlen és kérdőíves visszajelzésekből megerősítést kaptunk, hogy a motivációs céljaink – érdeklődés felkeltése, sokszínűség megmutatása – valóban megfelelően alakul mind a diákok, mind a tanárok esetében.

2.2. Feladat kiterjesztések

A versenyen használt feladatok többsége megállja a helyét egy interaktív vagy számítógép nélküli aktivitás során is. A motivációs, fejlesztő és ismeretgazdagító célokra ráerősíthet a játékosítás.

A kiterjesztett aktivitások tervezésénél fontos szempont, hogy több korosztály is sikerélménnyel gazdagodhasson, akár családi tevékenységként is végezhetőek legyenek. További célom volt, hogy mintát mutassak az egyes feladatok felhasználhatóságára – akár a tanórákon, akár azokon kívüli és motiváljam a közoktatásban dolgozókat a hód célok között megfogalmazott további felhasználásra [15].

Két aktivitást emelnék ki a többi közül. Mindkettőben a kiválasztott feladatok játékosítása fizikai megvalósítással történt: azaz a feladatok megoldásához valamilyen fizikai tevékenységre van szükség. Súlyok pakolása, kódtárca tekerése, hamburger, puzzle összerakása, kártyák tologatása, élőtársas, ... mind előfordul a megvalósítások között. Az előkészített feladatok nyomtatott, laminált papírokkal, gyurmával, horgolt eszközökkel – azaz bárki által elkészíthető alapokkal, olcsón beszerezhető eszközökkel készülnek, készültek el.

Az első aktivitás egy kalóz kincsesládájának a kinyitása, ahol négy feladat megoldása során összegyűjtött kóddal egy kincsesláda nyitható. A négy feladat összeválogatásánál figyeltem arra, hogy mind könnyebb, mind nehezebb feladatok szerepeljenek – de a megoldáshoz akár kisebb segítségekkel pár perc alatt mindenki el tudja jutni. A megoldás során a diákok, családok akár párhuzamosan is megválaszolhatják a kérdéseket, illetve fontos momentum, hogy akár meg is beszélhetik azokat. Koordinátornak egy ember is elegendő – aki az éppen megakadó résztvevőknek segít, illetve az egyes feladatok eszközeit alapállapotba visszaállítja.



Ezt az aktivitást teszteltem a Jövő utcájában családos programként, az ELTE IK nyílt napjain az érdeklődő végzős középiskolások körében, Kutatók Éjszakáján is.

A másik projekt egy akadályverseny, ahol az egyes állomásokon kell hód feladatokat megoldani, s így minél több pecsétet szerevezve eljutni a célig. Az állomások kidolgozásában az ELTE IK tanárszakos hallgatói vettek, vesznek rész – folyamatosan bővítve a „bevethető” feladatok tárházát. A legközelebbi akadályversenyt a 10. hód verseny díjkiosztójára tervezzük megvalósítani.

2.3. Programozók tesztelése

Az ELTE Informatikai Karának első éves alapoató kurzusa a Programozás, illetve az angol nyelvű Computer Science BSc képzésében a Programming elnevezésű kurzus, mely programozás-módszertani bevezetesként szolgál, célja az alapvető problémamegoldó algoritmusok megismerése és használata különböző egyszerű és összetettebb adatszerkezetekkel. Olyan általános sémákat igyekszik a hallgatók számára elérhetővé tenni, amely segítheti őket a problémák megértésében, a problémák részekre bontásában, az algoritmusok megalkotásában, majd végül a programok elkészítésében.

A korábbi évek tapasztalata alapján a hallgatók – s kiemelten az angol nyelvű BSc képzésben résztvevő hallgatók egy részének már az egyszerűbb algoritmusok megértése, felhasználása, majd később nehezebb feladatokban történő alkalmazása is problémákat okoz. A bonyolultabb algoritmusok, komplexebb problémák pedig óriási kihívást jelentenek számukra.

Az egyetemünkre jelentkező és angol nyelvű BSc képzésre felvételt nyert hallgatóknak egy matematikai és egy angol nyelvi tesztet kell megírniuk, és akik ezt a tesztet nem teljesítik megfelelő szinten, azok egy úgynevezett előkészítő féléven kezdenek meg tanulmányaikat. A Programming kurzus tapasztalatai alapján 2019 szeptembere óta már egy algoritmikus gondolkodás teszttel is bővült az év eleji felmérés, és 2020 szeptemberétől a magyar nyelvű hallgatók csoportbeosztását is egy hasonló teszttel segítjük.

A tesztek kidolgozásánál cél volt az, hogy az algoritmikus gondolkodás különböző szintjeit mérje, és ne támaszkodjon programozói előismeretekre. A differenciáló erő viszont megmutassa azt a határt, ami alatt a képzésbe lépőknek fejlesztési támogatást nyújthatunk egy előkészítő kurzuson.

Ehhez a teszt első verziójához [17] háttérkérdőívet is készítettünk, melynek önbevalláson alapuló eredményeit összevetettük a Programming képzés során nyújtott teljesítményekkel.

A tesztben szereplő kérdések a kiterjesztett Bloom-taxonómia alkalmazás, analízis, szintézis és alkotás szintjeit vizsgálták egy-egy könnyű, két-két közepes és egy-egy nehéz feladattal. A felidézési szint a kezdeti célok miatt – programozói előismeretek nélkül is megoldható –, és mivel a programozó képzésnek nem előfeltétele valamilyen programozási környezet vagy programozási nyelv ismerete, nem került tesztelésre.

Az eredmények előzetes feldolgozása alapján kijelenthetjük, hogy a teszt felülről megfelelő differenciáló erővel bír – azok kerültek az angol nyelvű képzésben előkészítő félévre, akiknek szükségük volt az erősítésre. Folyamatban van annak elemzése, hogy a teszten mutatott teljesítmény és a Programming, illetve Programozás kurzus elvégzésének nehézsége között fennáll-e valamilyen összefüggés. Ehhez háttérkérdőívvel kiegészítve a kurzusokon nyújtott teljesítményt vizsgáljuk. Az eredmények tekintetében fogalmazhatjuk meg majd azt, hogy a teszten első évre bejutó hallgatóknak nem lett volna szüksége a bevezető támogatásra, azaz alulról is megfelelő a teszt differenciáló ereje.

2.4. Egyetemi fejlesztő kurzus

2017-ben kezdtem el kidolgozni az első éves BSc programozó, angol nyelvű képzésbe bekerülők felzárkóztató kurzusát [18], mely az algoritmikus gondolkodás fejlesztésén kívül a csoportmunka, kooperáció, a kommunikáció és problémamegoldás területén is igyekezett erősíteni a résztvevőket.

A kurzus célját a következőképpen fogalmaztam meg a résztvevők számára:

- az algoritmikus gondolkodás és a problémamegoldási alapok biztosítása;
- annak megmutatása, hogy a problémamegoldás területén milyen szerepet játszhat az informatika (computer science);
- segíteni egyszerűbb célok megvalósítására készülő kisebb alkalmazások, programok megtervezésében és megvalósításában (illetve ezen folyamatok elsajátításában).

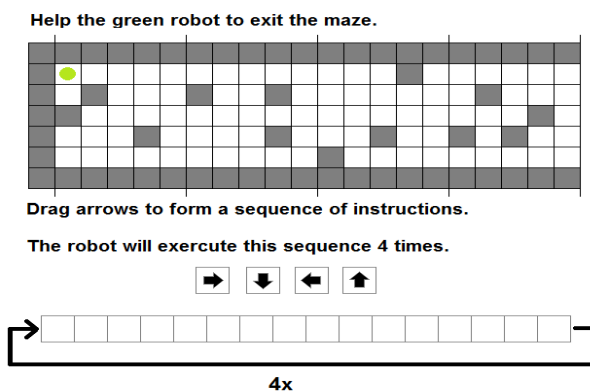
A kurzusban is helyet kapott a bebras kezdeményezés. A kurzus első moduljában („informatikai problémák és megoldásaik”) megfogalmazott céljaink:

- képet adni az informatikáról, számítástudományról;
- támogatni a hallgatókat abban, hogy elkezdjenek informatikai problémákról gondolkodni;
- helyet, időt biztosítani a problémamegoldásról, stratégiákról való beszélgetésekhez és a problémák, megoldási lépések megfogalmazásához.
- motiváltá tenni a hallgatókat a problémamegoldásra és az egyes megoldási stratégiákról való gondolkodásra.

Ehhez egy rövid problémamegoldással kapcsolatos bevezetőt követően a diákokkal hód feladatokat oldunk meg és értelmezünk. A hangsúly nem csupán a megoldáshoz való eljutáson van – illetve annak kielemezésén, hogy esetlegesen több út is elvezethet a megoldáshoz. A csoportmunka és megbeszélések a fent összefoglalt célokat is támogatják.

A kurzus második és harmadik moduljában az utasítások pontosságának fontosságát, a program struktúrákat sajátítják el a hallgatók még mindig programozási környezet, konkrét programnyelv használata nélkül. Itt többek között olyan hód feladatokat válogatunk össze, melyek ebben a témakörben adnak kihívásokat. Például a több nehézségi szinten megoldandó labirintus feladat (2. ábra), melyben

a hallgatóknak a minta megmutatásával segítséget nyújthatunk a megoldásban anélkül, hogy elárulnánk a helyes választ.



2. ábra: Labirintus feladat – közepes nehézségű szintje.

Az idei évben végez a második komolyabb csoport a kurzuson, így a következő szemeszter lezárásával már kiértékelhetjük a kurzus hatását a hallgatók programozói képzésben nyújtott teljesítményének függvényében.

Az első végzett csoport eredményeinek tekintetében megállapíthatjuk, hogy a kurzus során szerzett tudás beépült. A hallgatók többsége (91%) sikeresen vette az első féléves kurzusokat. Megfigyelhető volt, hogy nem csak biztosabb gondolkodási struktúrákkal rendelkeztek, de a tanuláshoz való hozzáállásuk (határidők betartása, előadások követése, felkészülés az órákra, csalások elkerülése, előadásokon való aktív részvétel) is pozitív változásokat mutatott, és képesek voltak kérdéseik pontos megfogalmazására is. Annak vizsgálata még hátra van, hogy mindezen attribútumokat mennyire és milyen szinten az adott kurzus, és mennyire a többi előkészítő kurzus (nyelvi, matematikai stb.), illetve az előkészítő félév látogatott kurzusainak összessége befolyásolta. A kurzus hatása mindenképpen érezhető az elsőéves tárgyak teljesítési eredményeinek tükrében.

2.5. Tanárképzés

Az ELTE IK tanárképzése nem csak az eszközhasználat és az általános pedagógiai módszertanok elsajátítását kívánja a hallgatóitól. Képzésünk fontos része az informatikai eszközökkel támogatott módszertani megoldások megismertetésén felül azok gyakorlati hasznosításának és megvalósulásának bemutatása. Ennek részeként több kurzusban is teret kapott az e-hód kezdeményezés megismerése, illetve mind a verseny feladatok elkészítésében, mind a kiegészítő aktivitásokban való részvétel.

3. Konklúzió

A bebras kezdeményezés megismerésével, Magyarországra hozásával és kiterjesztésével töltött 10 évem komoly elgondolások és kutatások alapját jelentik. A magyar megvalósulás immár nem csupán a versenyt jelenti, és nem csak a közoktatásban oktató pedagógusokat tudja segíteni, de az algoritmikus gondolkodás mérésére és fejlesztésére szűkített feladathalmaz a felsőoktatásban is megállja a helyét.

További feladatombnak tartom a verseny és a kiterjesztéseinek szélesebb körbe való eljuttatását, a pedagógusok, informatika tanárok támogatását mind a tanórákon, mind az azokon kívüli tevékenységeikben.

A felsőoktatásban kifejlesztett és használt tesztek, valamint a fejlesztést támogató kurzus folyamatos értékelésével, a hallgatók nyomonkövetésével igyekszem támogatni egyetemünk célkitűzését a lemorzsolódások csökkentésében és az oktatási színvonal megtartásában.

4. Köszönetnyilvánítás

A verseny és a hód kiterjesztések nem tudnának ilyen sikeresen helyt állni az ELTE IK tanárszakos hallgatóinak közreműködése nélkül. További köszönet illeti azokat a lelkes pedagógusokat, akik diákjaik részvételét lehetővé teszik, valamint a nemzetközi Bebras csapatnak, akik évről évre töretlenül munkálkodnak azon, hogy érdekesebbnél érdekesebb feladványok kerülhessenek terítékre.

Irodalom

1. J. M. Wing (2006) „*Computational Thinking*,” Commun. ACM, vol. 49, no. 3, pp. 33–35.
2. ISTE and CSTA, „*Operational Definition of Computational Thinking*,” (2011) Elérhető: <http://www.iste.org/docs/ctdocuments/computational-thinking-operational-definition-flyer.pdf>.
3. K. Brennan, M. Resnick (2012) „*New frameworks for studying and assessing the development of computational thinking*,” pp. 1–25.
4. Casey, P., J. (1997): „*Computer programming: A medium for teaching problem solving*,” New York: The Haworth Press. Computers in the Schools, XIII, 41–51.
5. OECD, 2010;
6. Chen-Chung, L., Yuan-Bang, C., Chia-Wen, H. (2011): „*The effect of simulation games on the learning of computational problem solving*,” Computers & Education, 57. 1907–1918.
7. Brennan, K, Resnick, M. (2012): „*New frameworks for studying and assessing the development of computational thinking*”, AREA.
8. Brennan, K. (2011): „*Creative computing: A design-based introduction to computational thinking*”, – elérhető: <http://scratched.media.mit.edu/sites/default/files/CurriculumGuide-v20110923.pdf> (utoljára megtekintve: 2016. 10. 25.)
9. Aiken, J. M., Caballero, M. D., Douglas, S. S., Burk, J. B., Scanlon, E. M, Thoms, B. és Schatz, M. F. (2012): „*Understanding Student Computational Thinking with Computational Modeling*”, PERC Proceedings.
10. Bell, T., Witten, I. H., Fellows, M. (2010): „*Computer Science Unplugged*”, – elérhető: <http://csunplugged.org/books> (utoljára megtekintve: 2016. 10. 25.)
11. CS4fun – elérhető: <http://www.cs4fun.org/> (utoljára megtekintve: 2020. 10. 20.)
12. Dagiene, V. (2006): „*Information technology contests – introduction to computer science in a attractive way*”, Informatics in Education, 5. 1.s., 37–46.
13. Cartelli, A., Dagiene, A., Futschek, G. (2010): „*Bebras Contest and Digital Competence Assessment: Analysis of Frameworks*”, International Journal of Digital Literacy and Digital Competence. Január–Március. 24-39.
15. Pluhár, Zs. (2016): „*Az informatikai gondolkodás és a hód*”, InfoDidact 2016: Informatika Szakmódszertani Konferencia, Zamárdi, Magyarország: Webdidaktika Alapítvány, (2016)
16. Pluhár, Zs. (2014): „*Az informatikai műveltség egyes dimenzióinak mérése*”, In: Buda András (szerk.) XIV. Országos Neveléstudományi Konferencia. Oktatás és nevelés – gyakorlat és tudomány: tartalmi összefoglalók., Debrecen, 472
17. Pluhár, Zs., Torma, H., Törley, G. (2018): „*Hallgatói teljesítményértékelés az algoritmikus gondolkodás tükrében*”, In: Szlávi, Péter; Zsakó, László (szerk.) InfoDidact, Budapest, Magyarország: Webdidaktika Alapítvány, 10 p.
18. Pluhár, Zs. Torma, H. (2019): „*Introduction to Computational Thinking for university students*”, In: Pozdniakov, Sergei N.; Dagiene, Valentina (szerk.) Informatics in Schools. New Ideas in School Informatics Cham, Svájc: Springer International Publishing, pp. 200-209., 10 p.