

Algoritmus gondolkodás fejlesztése a matematikaoktatásban

Improving algorithmic thinking in mathematics education

NAGY Zsuzsanna Timea, Dr. CSERNOCH Mária

Debreceni Egyetem Informatikai Kar
Kassai út 26 szám, Debrecen 4028
Telefon: +36 52 512 900, honlap: <https://www.inf.unideb.hu>
Email: nagy.zsuzsanna@inf.unideb.hu, csernoch.maria@inf.unideb.hu

Abstract

In recent years, our experience has shown that the majority of first-year students are missing the competences that are essential to develop algorithmic thinking and to apply these skills to computer / computer – assisted problem solving. The aim of this study is to develop algorithmic thinking and to integrate computer-based learning into mathematics education. There are several possible ways to do this, we focused on Sprego (Spreadsheet Lego) programming. The results of the research show that our teaching method developed the students' computational thinking and algorithmic skills, while they caught up with the basics of mathematics.

Keywords: Sprego, algorithmic thinking, mathematics, education

Kivonat

Az elmúlt években tapasztalataink szerint az elsőéves hallgatók többségénél hiányoznak azok a kompetenciák, amelyek elengedhetetlenek az algoritmus gondolkodás fejlesztéséhez és ezen képességek alkalmazása a számítógépes / számítógéppel támogatott probléma megoldásban. Jelen tanulmány célja az algoritmus gondolkodás fejlesztése, a számítógéppel való tanítás beépítése a matematikaoktatásba. Erre több lehetséges módszert találhatunk, mi a Sprego (Spreadsheet Lego) programozást helyeztük előtérbe. A kutatás eredményei mutatják, hogy a tanítási módszerünk fejlesztette a hallgatók számítógépes gondolkodását, algoritmikus készségüket, miközben felzárkóztak a matematika alapjaival.

Kulcsszavak: Sprego, algoritmikus gondolkodás, matematika, oktatás

1. BEVEZETÉS

A mai világban alapkövetelmény a számítógép ismeret, amely által lehetővé válik a mindennapok során felmerülő problémák gyors kezelése, megoldása. Alapvetően a középiskolában tanult matematikai fejezetek elengedhetetlenek a különböző egyetemi képzésekre való jelentkezés során. Hasonlóan fontos szerepet játszik az informatikai ismeretek megléte is, mint például a programozás, a mesterséges intelligencia, a matematikában a valószínűségszámítás és statisztika vagy éppen a közönséges differenciálegyenletekkel leírható hétköznapi modellek. Ezen témakörök mindenikében változatos formában fordulnak elő olyan különböző matematika fogalmak, mint függvények, sorozatok, tömbök, integrál- vagy differenciálszámítás.

A matematikai feladatok megoldásánál előnyt jelent a számítógépek használata, a különböző problémák megoldására alkalmazható szoftverek ismerete [1]. Ezen ismeretek folyamatosan fejleszthetők, illetve a további tanulásban és a mindennapi életben is felhasználhatók. Conrad Wolfram a számítógépes gondolkodás folyamatát a következő lépések megfogalmazásával írta le: kérdések meghatározása, kérdések átalakítása ahhoz, hogy tudjunk választ adni (számítógép segítségével), válaszok feldolgozása, eredmény értelmezése [2][3], amely lépések megegyeznek Pólya koncepció-alapú problémamegoldási megközelítésével [4]. Wing J. szerint az informatikus gondolkodás több absztrakciós szinten való gondolkodást jelent.

A problémamegoldás során gyakran találkozunk olyan feladatokkal, melyek megoldásához szükséges az adatokat táblázatokba [5] rendezni, ezután soronként vagy oszloponként műveletet végezni, például, összeadás, kivonás, százalékos arány. A táblázatkezelő programokat ma már összetettebb feladatok megoldására is használnak- mint például statisztikai feladatok megoldására, grafikonok készítésére,

egyszerűbb adatbázis kezelő műveletek megvalósítására. A táblázatkezelő program használata lehetővé teszi a hallgatók számára, hogy elmélyítsék tudásukat.

Az utóbbi években a matematika oktatás színvonala csökkent [6]. A diákok kognitív képességei- mint például a figyelem, emlékezés, de legfőképp a gondolkodás, melyhez szorosan kapcsolódik a problémamegoldás, illetve a tanulás, megváltoztak. A tapasztalat azt mutatja, hogy a mai követelményszint alacsonyabb [7][8], a diákok különböző szintű matematikai ismerettel kerülnek be az egyetemekre. Számos olyan alapvető fogalommal nincsenek tisztában, amelyek nélkülözhetetlenek a mérnök, informatika, matematika, fizika, kémia képzések esetén.

Kutatásom célja megvizsgálni miként lehet a matematikaoktatásba hatékonyan beépíteni a számítógéppel való tanítást. Arra a kérdésre kerestem a választ, hogy az alkalmazott módszer segítségével sikerült-e a hallgatók tudását megfelelő szintre emelni.

2. KUTATÁS

2.1. Teszt

A hiánnyal rendelkező elsőéves hallgatók számára a Sapientia EMTE marosvásárhelyi Karán bevezették a Felzárkóztató matematika tantárgyat, heti két óraszamban, amely keretén belül vizsgáltuk az elsőéves hallgatók fejlettségi szintjét a félév elején, illetve a félév végén.

A matematika következő ágait érintve állítottuk össze a feladatsort:

- Sorozatok generálása (számtani illetve mértani sorozatok)
- Számrendszerek, számjegyek kiírása
- Euklideszi algoritmus
- Függvények megoldása (elsőfokú, másodfokú, tört, trigonometrikus)
- Integrálás

A felsorolt fejezeteket eltérő módszerrel tanítottuk a hallgatók számára, mint a hagyományos oktatásban. Erősítettük az egyes fogalmak jelentésének a megértését, rávezetve a hallgatót a megoldás logikus algoritmusának felépítésére, majd ezután tértünk rá a komplexebb feladatok megoldására.

A fejezeteket egymásra építettük fel, egy logikus sorrendet állítottunk össze a feladatok megoldására, folyamatosan rávezetve a hallgatót a Conrad Wolfram által megfogalmazott számítógéppel támogatott problémamegoldás folyamatának használatára. A következő alkalomkor mindig erősítettük az előző órán vett fogalmakat és ellenőriztük, hogy megtörtént-e a megértés.

A fenti fejezeteket táblázatkezelő segítségével szemléltettük a hallgatókkal, Sprego programozást felhasználva[9][10].

A táblázatkezelő használata mellett még arra hívtuk fel a hallgatók figyelmét, hogy minden egyes feladatot próbáljanak az eddig tanultak alapján megközelíteni, ábrázolják az illető függvényt, mert úgy könnyebb lesz a feladat megoldása is.

2.2. Feladatok

A félév elején és végén írtunk egy - egy tesztet a hallgatókkal, melyet Google űrlap segítségével készítettük el. A feladatsor tíz feladatot tartalmazott, a maximálisan elérhető pontszám 95 pont volt.

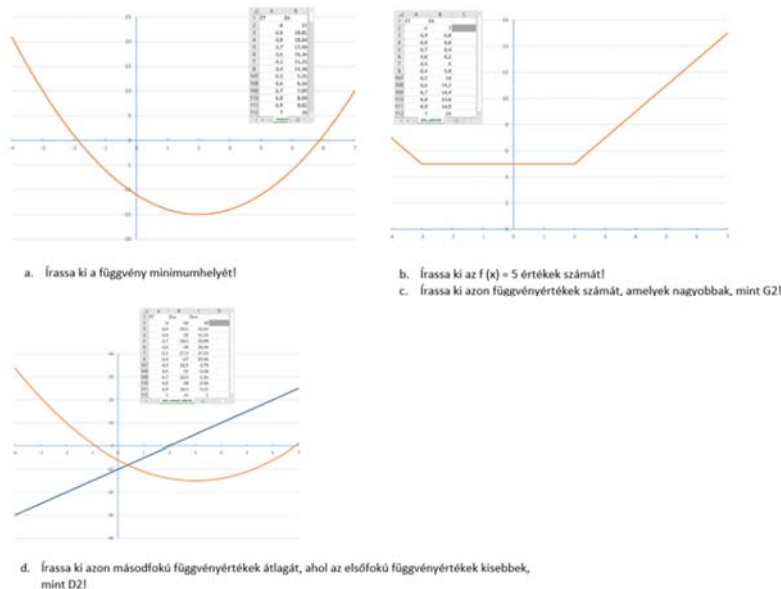
A feladatokat úgy állítottuk össze, hogy két nagy csoportba legyenek sorolhatók: az első csoportba tartozó feladatok (Ábra 1) a mindennapi életből vett adatokat tartalmazta (ország, főváros, terület, lakosság). Például egy adathalmazból, amely a világ országaira vonatkozó adatokat tartalmazott, ki kellett írassak a legnagyobb területű ország fővárosát, vagy azon országok átlagos területét, amelyek lakossága kisebb, mint egy adott érték, stb. A táblázatkezeléssel kapcsolatos feladatok a TAaAS projektben (Testing Algorithmic and Application Skills) található meg [11], amelyek által mérhető a tanulók algoritmus és képlet felépítési (a-e. feladatok), illetve a képlet kiértékelő készségeik (f. feladat). Az a. feladat a lineáris keresés algoritmusát helyezi előtérbe. A b. feladat megoldása követeli a népsűrűség kiszámításának az ismeretét, illetve arra is kell itt figyeljenek a tanulók, hogy a népességi értékek ezerben vannak megadva. A c. és e. kérdés a feltételes számolásról szól, a d. feladat pedig egy feltételes átlagszámításról. Az első csoport utolsó feladata egy összetett képlet elemzését tartalmazza [12].

	A	B	C	D	E	F	G
1	ország	földrész	főváros	terület	lakosság (ezer)		
2	Afghanistan	Asia	Kabul	647500	27756		
3	Albania	Europe	Tirana	28748	3545		
4	Algeria	Africa	Algiers	2381740	32278		
5	American Samoa	Oceania	Pago Pago	199	69		
6	Andorra	Europe	Andorra la Vella	468	68		
7	Angola	Africa	Luanda	1246700	10593		
8	Anguilla	Amerika	The Valley	102	12		
234	Yugoslavia	Europe	Belgrade	102350	10657		
235	Zambia	Africa	Lusaka	752614	9959		
236	Zimbabwe	Africa	Harare	390580	11377		

- Írassa ki a legnagyobb területű ország fővárosát!
- Írassa ki az egyes országok népsűrűségét!
- Írassa ki az afrikai országok számát!
- Írassa ki azon országok átlagos területét, amelyek lakossága kisebb, mint G2!
- Írassa ki azon országok számát, amelyek területe nagyobb, mint G2!
- Mit csinál az alábbi összetett képlet?
{=SZUM(HA(B2:B236="Europe"; HA(BAL(A2:A236)="A";1)))}

Ábra 1. Első csoportba sorolt feladatok

A második csoport (Ábra 2) konkrét matematikai feladatokat tartalmazott, mint például egy függvény minimumhelyének meghatározása, vagy azon másodfokú függvényértékek átlaga, ahol az elsőfokú függvényértékek kisebbek, mint egy adott érték, stb.



Ábra 2. Második csoportba sorolt feladatok

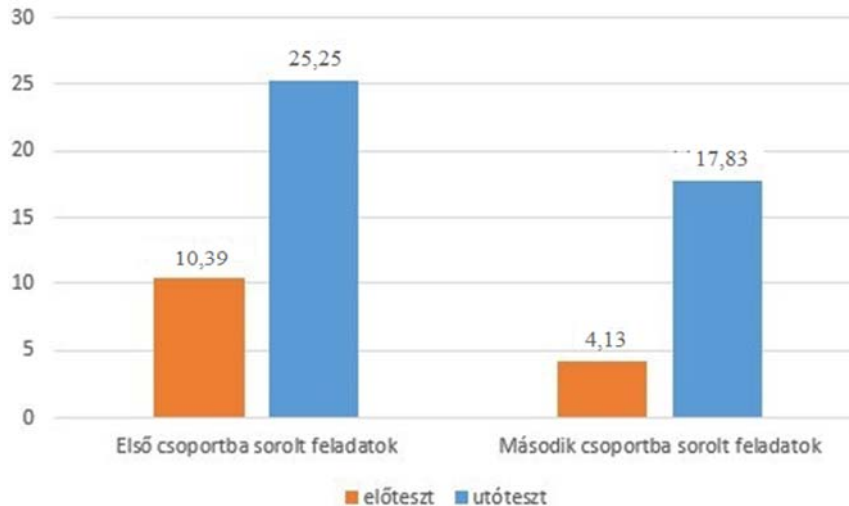
Mindkét csoport feladatai között voltak olyan feladatok is, amelyeknek a megoldási algoritmus megegyezett. Az a. feladat megoldása ebben a csoportban szorosan kötődik az első csoport a. feladatával, annyi eltéréssel, hogy itt minimumkeresést, míg az első csoportnál egy maximumkeresést kellett elvégezniük a diákoknak. A második csoport b. feladatának algoritmus megegyezik az első csoport c. feladat algoritmusával, a második csoport c. feladata az első csoport e. feladat algoritmusával, illetve a második csoport d. feladata az első csoport d. feladatának algoritmusával. Ennek a hasonlóságnak a célja az volt, hogy rávezessük a tanulókat a matematikai feladatok megoldására egy mindennapi életből vett adatokon elvégzett feladatok segítségével.

2.3. Minta

A kísérletre a 2020/2021 tanév első félévében került sor, 64 diákot teszteltünk különböző szakokról: informatika, számítástechnika, gépészmérnöki, mechatronika, távközlés és automatizálás szak.

3. EREDMÉNY

Az eredményeket először a két csoportba sorolt feladatok alapján vizsgáltuk meg. Az előteszt során az első csoportba sorolt feladatokat sikerült jobban megoldaniuk a hallgatóknak, mint a második csoport feladatait (Ábra 3). A mindennapi feladatok megoldásában jobban teljesítettek a hallgatók az utótesztnél is, arra következtethetünk, hogy a Sprego programozás segíti az életszerű problémák megoldásában. (A félév során a Sprego programozást felhasználva a konkrét matematikai feladatok megoldásában nagyobb fejlődést mutattak, sikerült megemelni a hallgatók általános tudásszintjét.)

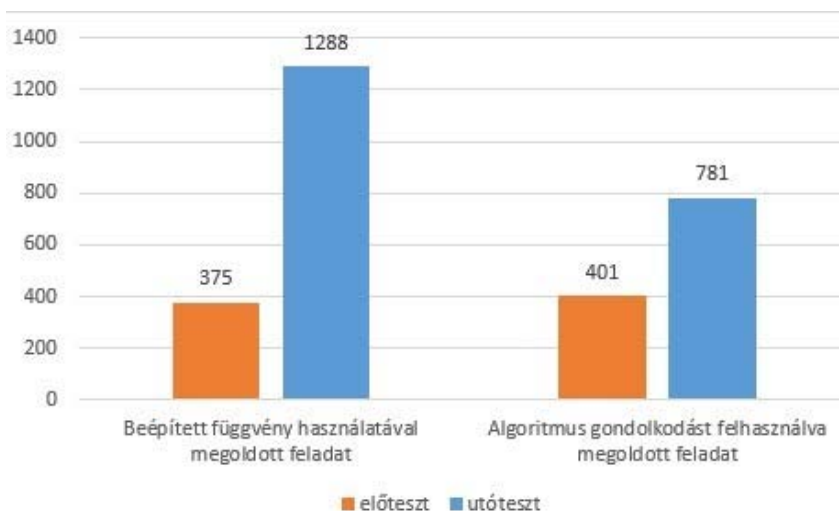


Ábra 3. Előteszt és utóteszt – a két csoportba sorolt feladatok átlagpontszámai

Az elért pontszámok arra engednek következtetni, hogy a matematikai feladatok megoldásában jobban fejlődtek a hallgatók, ami azt bizonyítja, hogy az általunk használt módszer segítette a felzárkóztatását a hallgatóknak.

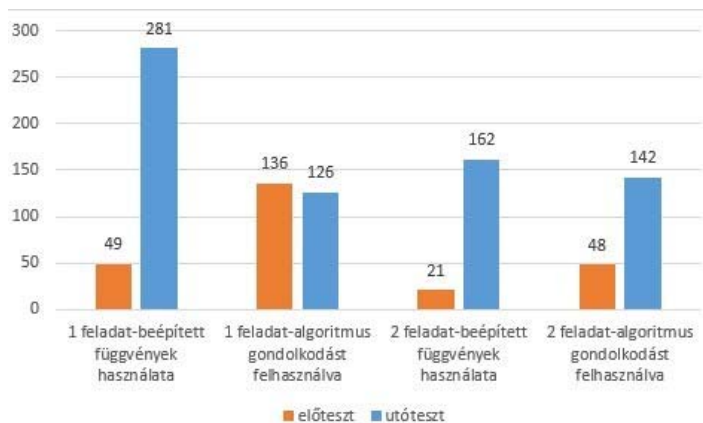
Amennyiben azokat a feladatokat vizsgáljuk, amelyeknek megoldása kétféleképpen történhetett, az egyik lehetséges megoldás beépített függvény használatát igényelte, a másik a függvények algoritmus felépítését, akkor a kapott adatokat az Ábra 4 szemlélteti.

Van fejlődés a hallgatók algoritmus gondolkodásában, viszont nagyobb a fejlődés a beépített függvény használatot igénylő feladatmegoldás elsajátításában. Míg az előteszt során ugyanolyan arányban használták mindkét megoldási stratégiát, az utóteszt adatai arra utalnak, hogy a feladatmegoldás során a hallgatók inkább a beépített függvények használatát léptették előnyben az algoritmus gondolkodást igénylő megoldás helyett. Mindez azzal is magyarázható, hogy a középiskolai oktatás során az informatikai tárgy keretén belül már találkoztak ezekkel a fogalmakkal, használták a feladatok megoldásában.



Ábra 4. Előteszt és utóteszt – elért pontszámok

Az alábbi ábrán (Ábra 5.) két olyan feladat eredményei láthatók, melyeknek a megoldási algoritmus megegyezik. Az 1 feladat: *Írassa ki azon országok átlagos területét, amelyek lakossága kisebb, mint egy adott érték*, a 2 feladat: *Írassa ki azon másodfokú függvényértékek átlagát, ahol az elsőfokú függvényértékek kisebbek, mint egy adott érték*. A kapott adatok arra utalnak, hogy a hallgatók inkább meg tudták oldani az 1 feladatot, mint a 2 feladatot. Az előtesztben kisebb arányban oldották az algoritmus gondolkodást, a félév végére nem sikerült látványos eredményt elérni, az online oktatást nem vált a javunkra, mivel online térben a feladatsor kitöltése során egyes hallgatók együttműködtek. Ezt a mérést még egyszer meg fogjuk ismételni.



Ábra 5. Az 1. és 2. feladatok eredményei

4. KÖVETKEZTETÉS

A kapott adatok értelmében, ha azt vesszük alapul, hogy hogyan fejlődtek a hallgatók a félév elején és a végén, akkor arra a következtetésre juthatunk, hogy az utóteszt megoldásában jobban teljesítettek, mint az előteszt során. Megállapíthatjuk, hogy a diákokat segítette a számítógéppel támogatott matematikaoktatás.

Ezt a felmérést viszont a következő tanév hallgatóival is el szeretnénk végezni. A kapott eredményeket jobbra szeretnénk tenni és a hangsúlyt inkább az algoritmus gondolkodás fejlesztésére tennénk.

IRODALMI HIVATKOZÁSOK

- [1] Wolfram C.: *Stop teaching calculating, start teaching math*, https://files.wolframcdn.com/pub/www.computerbasedmath.org/Education_talk_transcript.pdf (Utolsó letöltés: 2022. 09.15).
- [2] Wolfram C., *The computational thinking process*, <https://www.computerbasedmath.org/computational-thinking-process-poster/> (Utolsó megnézés: 2022. 09.15).
- [3] Wolfram C., *The Math(s) Fix: An Education Blueprint for the AI Age*. Wolfram Media, Inc. 2020.
- [4] Pólya G., *How To Solve It. A New Aspect of Mathematical Method*. (2nd Edition 1957), 1954, Princeton, NJ: Princeton University Press.
- [5] Jones K. Using Spreadsheets in the Teaching and Learning of Mathematics: a research bibliography. *MicroMath*, 2005, 21(1), 30-31.
- [6] *International student assessment (pisa)*, https://www.oecd-ilibrary.org/education/mathematics-performance-pisa/indicator/english_04711c74-en (Utolsó megnézés: 2022. 09.15).
- [7] *Érettségi tételek 2007-től 2020-ig Romániában*, <https://www.pro-matematica.ro/bacalaureat/2007.php> (Utolsó megnézés: 2022. 09.15).
- [8] *Oktatási Minisztérium – Románia*, <https://edu.ro> (Utolsó megnézés: 2022. 09.15).
- [9] Csernoch M., Biró P., *Sprego programming*. *Spreadsheets in Education*, 2015, 8(1), 1-38.
- [10] Csernoch M., *Programozás táblázatkezelő függvényekkel – Sprego*, 2014, Műszaki Könyvkiadó: Budapest, Hungary.
- [11] Csernoch M., Biró P., Máth J., Abari K., *Testing Algorithmic Skills in Traditional and Non-Traditional Programming Environments*. *Informatics in Education*. 14.175-197.10.153888/infedu.2015.11.
- [12] Csapó G., Sebestyén K., Csernoch M., Abari K. *Case study: Developing long-term knowledge with Sprego*. *Education and Information Technologies*, 2021, 26.10.1007/s10639-020-10295-0.
- [13] Wing J. M., *Computational Thinking*. *Communications of the ACM*, 2006, Vol.49, No.3.