

# Szövegértés és algoritmikus gondolkodás

Kátai Zoltán

katai\_zoltan@ms.sapientia.ro  
SAPIENTIA-EMTE

**Absztrakt.** A számítógépes gondolkodás készségének kialakítása végett egyre nagyobb hangsúlyt kap, egyre több tudományterületen, a programozás oktatás. Egy friss kutatásban [6] a szerzők azt vizsgálták, hogy 7 tapasztalt kolumbiai egyetemi oktató és 44 hallgató mely készségeket és milyen előismeretet tekint fontos előfeltételnek a hatékony programozás-oktatáshoz. Messze legfontosabb, úgy tanárok, mint hallgatók, a fejlett szövegértési készséget jelölték meg. Másfelől, a vizsgálatok alátámasztották, hogy a kódolási készség fejleszthető kódolvasás révén [8,9]. Az általános szövegértés és a kódolvasás között helyezhető el az, ha egy mindennapi szövegben (akár irodalmiban is) algoritmikus elemeket kell beazonosítani, értelmezni. Ebben a tanulmányban azt vizsgáljuk, hogy az első éves hallgató programozó hallgatók szövegértési készsége technikai szövegen, valamint az 1. és 2. szintű algoritmikus gondolkodási készsége [4] számítógép-független környezetben (algoritmus-felismerés általános szövegben, algoritmus-végrehajtás matematikai feladaton), miként harmonizál.

**Kulcsszavak:** szövegértés, algoritmikus gondolkodás, programozás-oktatás

## 1. Bevezetés

Az algoritmus fogalma messze visszanyúlik a történelembe, és nyilván nem kapcsolódik kizárólagosan a számítógépek világához. Kiváló példa e tekintetben Euklidesz több évezredes matematikai algoritmusai két szám legnagyobb közös osztójának kiszámítására. Másfelől viszont vitathatatlan, hogy a számítógépek térhódítása hozzájárult ahhoz, hogy az algoritmikus gondolkodás fontossága ennyire előtérbe került a XX. század második felétől kezdődően. Sőt néhány éve szárnyra kapott a számítógépes gondolkodás [1] fogalma is, amit egyes szerzők a negyedik alapvető készségnek tekintenek (az olvasás, írás és aritmetika mellett), amely nélkülözhetetlen a XXI. század embere számára [2,3].

Kétségtelen, hogy az újonnan megalkotott számítógépes gondolkodás fogalma leginkább az algoritmikus gondolkodás fogalmához áll közel (bár a két fogalom nem fedi egymást; az utóbbi, ahogy a neve is jelzi, sokkal közvetlenebbül kapcsolódik a számítógépek programozásának készségéhez), bizonyos mértékig ebből nőtte ki magát, ebben gyökerezik. Ebből adódóan a számítógépes gondolkodás hangsúlyozása a modern társadalomban, továbbra is előtérben tartja az algoritmikus gondolkodás fontosságát is. Lévéen szó egy összetett készségről. Szlávi és Zsakó [4] az algoritmikus gondolkodás fogalmát a következő szintekre bontja le: (1) felismerés–megértés, (2) végrehajtás, (3) elemzés, (4) alkotás, (5) megvalósítás, (6) módosítás–átalakítás, (7) (komplex)tervezés.

A számítógépes gondolkodás készségének kialakítása végett egyre nagyobb hangsúlyt kap, egyre több tudományterületen, a programozás oktatás. Bár e megállapítás elsősorban a felsőoktatásra vonatkozik, egyre többen hangsúlyozzák a programozás-oktatás fontosságát már középiskolai szinten, sőt léteznek kezdeményezések már elemi szinten is. Másfelől a több évtizedes tapasztalat azt mutatja, hogy igazi kihívást jelent hatékonyan programozást oktatni, akár egyetemi hallgatóknak is. Számos kutatás vizsgálta ennek okait, módszereit, illetve azt, hogy mi lehet a titka, előfeltétele az eredményességnek úgy oktatáspolitikai, oktatói, mint tanulói oldalról. A

British Computer Society [5] egyik tanulmánya feltárta, hogy egyik fő ok, amiért a felsőoktatási tanulmányi programok Programozás-I tantárgyai nem hozzák a várt eredményeket az, hogy a programozás-oktatásnak –más tantárgyakkal ellentétben– középiskolai szinten nincsen összehangolt, szisztematikus előkészítése. Továbbá, az hogy egyes líceumok, bizonyos szakirányain létezik valamennyi programozás-oktatás, azt eredményezi, hogy rendkívül heterogén csoportoknak kell a Programozás-I tantárgyakat oktatni.

Egy friss kutatásban [6] a szerzők azt vizsgálták, hogy 7 tapasztalt kolumbiai egyetemi oktató és 44 hallgató mely készségeket és milyen előismeretet tekint fontos előfeltételnek a hatékony programozás-oktatáshoz. A szakirodalommal összhangban [7,8,9,10] a vizsgáldóság 4 kategóriából ölelt fel potenciális tényezőket: feladatmegoldás, programozási előismeretek, programozói készségek, algoritmikus gondolkodás. Az eredmények azt mutatták, hogy dominánsan a készségekben rejlik a siker titka, és nem annyira az előismeretekben vagy előzetes tapasztalatban. Messze legfontosabbnak, úgy tanárok, mint hallgatók, a fejlett szövegértési készséget jelölték meg. Mindezekkel összhangban a jelen kutatásban azt vizsgáltuk, hogy különböző háttérű (reál/humán, elméleti/szak-liceumi, programozói:igen/nem előképzésű) I. éves hallgatók szövegértési és algoritmikus gondolkodási készsége miként cseng össze.

### **1.1. Szövegértés vs. algoritmikus gondolkodás**

Az olvasás a leírtak által vezetett gondolkodás. Az olvasási folyamat két fő részre osztható. Az első rész a dekódolás, vagyis a vizuális élmény alapján a betűsorok megfejtése, a betűknek a megfelelő beszédhangokkal történő megfeleltetése. A második részben történik a megértés és a jelentés azonosítása. Megkülönböztetünk ismeretszerző olvasást, tanulási célú olvasást, élményszerző olvasást, kereső olvasást, áttekintő olvasást, feladatonosító olvasást, javító vagy ellenőrző olvasást, korrektúraolvasást, stb. [11]

Az értelmes, elemző olvasás (értő olvasás) a mindennapi élet szükségszerűsége. A szövegértés azt feltételezi, hogy minden szó értelemmel bír az olvasó részére, hogy ezek jelentését be tudja építeni az egész szöveg mondanivalójába. Egy olvasmány logikai felépítése sokféle lehet: témaszerinti, időrendi, ok-okozat, problémamegoldás stb. elrendezés. Az értő olvasó képes kiemelni egy szövegből a lényegét, és átlátja az összefüggéseket: miként kerül kifejtésre a téma a főpontok révén, hogyan támogatják az alpontok a főpontokat. A jó olvasó kapcsolatot keres meglévő tudása és a szöveg új információi között, meghatározza, hogy melyek a szöveg legfontosabb témái, gondolatai, következtetéseket von le, új gondolatokat fogalmaz meg, összefoglalja, újraalkotja az olvasottakat [12]. Továbbá, a hatékony szövegértés kéz-a-kézben jár az eredményes szövegalkotással (a jó írók, egyben jó olvasók is).

Az értő olvasás számos alapelve kamatoztatható a hatékony számítógépes kódolvasás fejlesztésében is. A számítógépes programok is karakterekből, kulcsszavakból, egyszerű és összetett mondatokból (utasítások) állnak. Nem nehéz párhuzamot találni a bekezdés, fejezet, alfejezet fogalmaknak is. Vizsgálatok alátámasztották, hogy a kódolási készség fejleszthető kódolvasás révén [8,9]. Az általános szövegértés és a kódolvasás között helyezhető el az, ha egy mindennapi szövegben (akár irodalmiban is) algoritmikus elemeket kell beazonosítani, értelmezni.

Ebben a tanulmányban azt mutatjuk be, hogy az első éves haladó programozói csoport esetében miként harmonizál a szövegértési készség technikai szövegen, és az 1. és 2. szintű algoritmikus gondolkodási készség [4] számítógép-független környezetben (algoritmus-felismerés általános szövegben, algoritmus-végrehajtás matematikai feladaton).

## 2. Kísérlet

A Sapientia-EMTE Marosvásárhelyi Karán az informatika és mérnöki szakos (számítástechnika, automatizálás, távközlés, mechatronika, gépészmérnöki) reál-hallgatók tanulnak kötelezően programozást már az első félévtől. Ez közel 200 hallgatót jelent rendkívül sokszínű középiskolai háttérrel. Egyesek 4 évet intenzív informatika szakon, mások pedig matematika-informatika szakon tanultak programozást. Akadnak olyanok is, akik természettudományok szakon két év alatt kaptak programozás-ízeltőt. A legtöbben semmilyen középiskolai előismerettel nem rendelkeznek a programozás területéről, de néhányan közülük autodidakta módon belekóstolt a programozásba.

Minden tanév elején tesztet íratunk az elsőéves informatika és mérnöki szakos hallgatókkal, hogy feltérképezzük a programozói, illetve algoritmikus gondolkodási készségeiket. Általában 3 feladatot kell megoldaniuk a diákoknak: egy egyszerűbb (F1.1) és egy bonyolultabb (F1.2) programozási feladatot, bármely programozási nyelven, valamint egy programozás-mentes logikus- és algoritmikus gondolkodást tesztelő feladatot (F1.3). Az 1. táblázatban a jelen tanulmányban feldolgozott 25 haladó programozó (24 fiú, 1 lány; 16 informatika, 4 számítástechnika, 4 automatizálás, 1 távközlés) e tesztbeli eredményei láthatók.

Az idén (2015-16 tanév) egy második teszt révén azt is vizsgáltuk, hogy milyen a hallgatók szövegértelmezési készsége technikai szövegeken (F2.1), valamint hogy képesek-e algoritmusokat kiolvasni általános szövegekből (F2.2). A teszt azt is mérte, hogy képesek-e végrehajtani egy algoritmust jól ismert matematikai kontextusban (F2.3) (lásd a Függelék).

1. teszt			
Feladatok	F1.1	F1.2	F1.3
Maximális pontszám	10	10	10
Elért pontszámok átlaga	8,26	7,15	5,36
Átlag-eljesítmény %-ban	83	71	54

**1. táblázat:** A haladó csoport eredményei a kezdő/haladó csoportokba való szétosztást célzó teszten.

## 3. Eredmények

A 2. táblázat bemutatja, hogy milyen eredményeket értek el a haladó programozó hallgatók a szövegértés feladat kérdéseinek megválaszolásánál. Megfigyelhető, hogy az összeredményt jelentősen lehúzták a nyelvi jellemzőkre rákérdező 14. és 15. kérdésekre adott válaszok.

2.teszt (F2.1)																	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	Össz
1	5	5	5	5	5	5	5	5	10	5	5	5	5	9	10	10	100
1	4,3	2,7	4,3	3,4	4,3	4,5	4,5	4	5,8	4,8	3,5	3,3	0,7	2	8,6	8,4	70
100	86	54	85	69	86	90	90	80	58	96	70	66	14	22	86	84	70

**2. táblázat:** A szövegértési feladat eredményei (a 17 kérdéshez tartozó maximális pontszámok, elért pontszámok átlagai és átlagteljesítmények %-ban).

Az F2.2a feladat azt célozta meg, hogy képes-e érzékelni a diák egymásba ágyazott feltételes szerkezetet (ha-szerkezetet) egy általános szövegben. Az F2.2b feladat azt tesztelte, hogy észrevesznek-e a hallgatók szekvencia, illetve elől-tesztelés ismétlési szerkezetet egy irodalmi szövegben.

Az eredmények azt mutatják, hogy a hallgatók jól érzékelték az egymásba ágyazott feltétel-rendszert (F2.2a.1 részfeladat), de gondjaik voltak ott, ahol a szövegből ki kellett következtetni (nem volt közvetlenül benne a szövegben) a megfelelő döntési ághoz tartozó „jóslatot”. Amíg a belső ha-szerkezet hamis ágához a hallgatók 82%-a helyesen a „nem lenne sok haszna” jóslatot társította, addig a külső ha-szerkezet hamis ágához sokan csak annyit írtak, hogy „*férfi lesz a Dalai láma*”. A megfelelő 2. kérdésre a hallgatók csupán 27%-a válaszolta azt, hogy „nem lesz jelentős változás”, vagy adott ezzel egyenértékű választ. Megjegyzendő viszont, hogy e pontatlanság nem a ha-szerkezet helytelen értelmezésére vonatkozott.

Az F2.2b feladat eredményei azt mutatják, a diákok 84%-ban jól érzékelték a szekvencia-struktúrát (1-2. részfeladatok). A ciklusfeltétel esetében (3. részfeladat) csak 38% válaszolta helyesen azt, hogy „*távol van a nagynéni*” (vagy adott ezzel egyenértékű feltételt). A legtöbben azt válaszolták, hogy „*van még feladat a kertben*”. Itt megint csak kijelenthető, hogy a hallgatók helyesen érzékelték az ismétlési-szerkezetet, csak hogy figyelmetlenek voltak arra a „részletre” nézve, hogy a nagynéni előírta munkamenet algoritmizálásáról szolt a feladat, nem pedig a Panka által végrehajtottól.

2.teszt (F2.2)								
Feladatok	F2.2a			F2.2b				Össz
Részfeladatok	1	2	3	1	2	3	4	
Maximális pontszám	20	15	15	10	10	20	10	100
Elért pontszámok átlaga	19	4	12	8,4	8,4	7,6	8	67
Átlag-eljesítmény %-ban	93	27	82	84	84	38	80	67

**3. táblázat:** A algoritmus-elemeket detektálódó feladat eredményei.

Mivel az A-B, illetve B-A részfeladatok természetükben identikusak voltak, ezért a B-A esetben való aluteljesítést elsősorban figyelmen kívül tudhatjuk be. Amint a 4. táblázatból is látható a hallgatók, ahogy várható is volt, jól teljesítettek az algoritmus-végrehajtási készséget tesztelő feladatokon. A csillaggal megjelölt alacsony teljesítmény az eredmények értelmezésére vonatkozik. Csak 40%-ban állt össze a hallgatókban az összkép, hogy szorzási algoritmusról van szó. Megjegyzendő, hogy a diákok kaptak 5 részpontot, amennyiben kihangsúlyozták, hogy az A-B és B-A „algoritmusok” identikus eredményhez vezetnek.

2.teszt (F2.3)										
Részfeladatok	A - B				B - A				*	Össz
	1	2	3	4	1	2	3	4		
Maximális pontszám	10	10	5	15	10	10	5	15	20	100
Elért pontszámok átlaga	9,6	9,2	4,3	12	8,4	7,2	3,2	8,7	8	70
Átlag-eljesítmény %-ban	96	92	86	77	84	72	64	58	40	70

**4. táblázat:** A algoritmus-végrehajtási feladat eredményei.

A második teszt 1, 2, és 3. feladataira kapott 70, 67, 70 százalékos átlagteljesítményeket összehasonlítottuk, kettőnként, t-próba segítségével is, és, ahogy várható is volt, nem találtunk szignifikáns különbségeket közöttük.

## 4. Következtetések

A dolgozatban bemutatott vizsgálat konfirmálta, hogy a jó programozói készség mögött fejlett szövegértési készség húzódik meg, illetve hogy megfelelő algoritmikus gondolkodási készség társul hozzá. Az eredmények arra engednek következtetni, hogy a megfelelő szövegértési készség valóban előnyt jelent az eredményes algoritmikusgondolkodás-fejlesztési törekvésekhez.

A kutatás következő lépése, hogy feldolgozzuk a többi humán és kertész szakos elsőéves hallgató (humán, kertész, reál-kezdőprogramozók) teszteredményeit is, és összehasonlító elemzést végezzünk különböző szakirányú, valamint más-más elő-felkészültséggel rendelkező csoportokra vonatkozóan.

## Irodalom

1. Denning, P. (2009). The profession of IT- Beyond computational thinking. Communications of the ACM, 52 (6), 28-30.
2. CSTB, Computer Science Telecommunications Board, (2010). Report of a workshop on the scope and nature of computational thinking. Washington, D.C.: The National Academies Press, [http://www.nap.edu/openbook.php?record\\_id=12840&page=R1](http://www.nap.edu/openbook.php?record_id=12840&page=R1)
3. Qualls, J. A., & Sherrell, L. B. (2010). Why computational thinking should be integrated into the curriculum. Journal of Computer Science in Colleges, 25, 66-71.
4. P Szlávi, L Zsakó. (2014). Key concepts in informatics: algorithm. Acta Didactica Napocensia. Volume 7, Number 1.
5. A. McGettrick et al., Grand Challenges in Computing Education, London, United Kingdom: The British Computer Society, 2004.
6. Prados A. V., Rivera L. R., Gerardo Sarria. 2015. Competences that facilitate the achievement of the objectives of an introductory programming course. Proceedings of 18<sup>th</sup> International Conference on Interactive Collaborative Learning, World Engineering Education Forum, Florence, Italy.
7. R. Lister et al., A Multi-National Study of Reading and Tracing Skills in Novice Programmers., New York: Conference on Innovation and Technology in Computer Science Education., 2004.
8. M. Lopez, J. Whalley, P. Robbins and R. Lister, Relationships Between Reading, Tracing and Writing Skills in Introductory Programming, Sydney, Australia: ACM, 2008.

9. R. M. Kaplan, "Teaching Novice Programmers Programming Wisdom.," in 22<sup>nd</sup> Annual Psychology of Programming Interest Group, Madrid, España, 2010, pp. 112-120.
10. Z. Karimi and S. Wagner, The Influence of Personality on Computer Programming: A Summary of a Systematic Literatur Review, Stuttgart: Universitat Stuttgart, 2014.
11. Gósy Mária. A szövegértő olvasás. 2008. <http://www.anyanyelv-pedagogia.hu/cikkek.php?id=25>
12. Dr. Antal Sándor. Szövegértés és szövegértő olvasás. 2010. <http://extensii.ubbcluj.ro/odorheiusec/tanulutmutat-tav/4%20felev/Szovert%20TAV%202010.pdf>

## Függelék

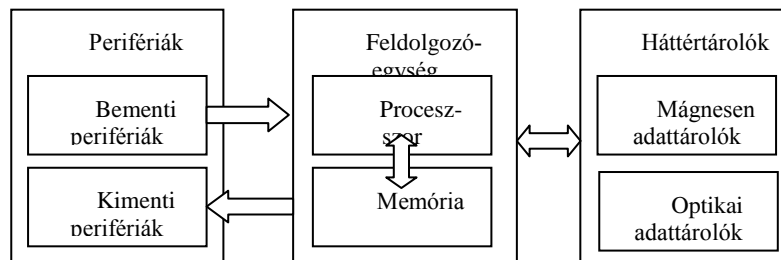
### 1. FELADAT

#### *A számítógép logikai felépítése*

*A mai értelemben vett számítógépek működési elveit a haditechnikában megszerzett tapasztalatok felhasználásával Neumann János, magyar származású tudós dolgozta ki 1946-ban. Az alapelvek szerint a számítógép olyan matematikai problémák megoldására szolgál, amelyekre az ember önállóan is képes lenne. A cél a műveletek végrehajtási idejének meggyorsítása. Ennek érdekében minden feladatot összeadások sorozatára kell egyszerűsíteni, ezután következhet a számolás mechanizálása. Az elmúlt évtizedekben a számítógépek nagyfokú fejlődésen mentek keresztül, de elvi felépítésük nem változott.*

*Ahhoz, hogy jobban megértsük a számítógép működését, ismernünk kell a számítógép logikai felépítését, valamint a különböző logikai részek szerepét.*

*A számítógép részeit három fő csoportba sorolhatjuk. A feldolgozóegység a számítógép lelkének tekinthető, két legfontosabb része a processzor és a memória. A processzor értelmezi és végrehajtja az utasításokat, a memória az programok és adatok tárolását teszi lehetővé. A programmemóriában található a programot alkotó utasítássorozat. Az adatmemóriában találhatók a feladat megoldásához szükséges kezdeti adatok, a program futtatása során keletkezett részeredmények, valamint a program befejezése utáni végeredmények. A memórián kívüli tárolóeszközök a háttértárolók, amelyek elsődleges feladata az adatok hosszú távú megőrzése a számítógép kikapcsolása után is. Tárolási elveik alapján léteznek mágneses és optikai háttértárolók. A legfontosabb háttértárolók a merevlemez és az író/olvasó egységek. A perifériák teremtik meg a kapcsolatot a számítógép és a felhasználók között. A számítógép központi egységéhez kívülről csatlakozó eszközök, az adatok ki- vagy bevitelét teszik lehetővé. Működésük alapján két csoportra oszthatók, bemeneti és kimeneti perifériákra. A bemeneti perifériák segítségével adatokat és utasításokat vihetünk be a számítógépbe, a kimeneti perifériákon keresztül a gép közli a műveletek eredményeit, a kimeneti adatokat. Legfontosabb bemeneti perifériák a billentyűzet és az egér, legfontosabb kimeneti perifériák a képernyő és a nyomtató.*



**1. Feladatlap:**

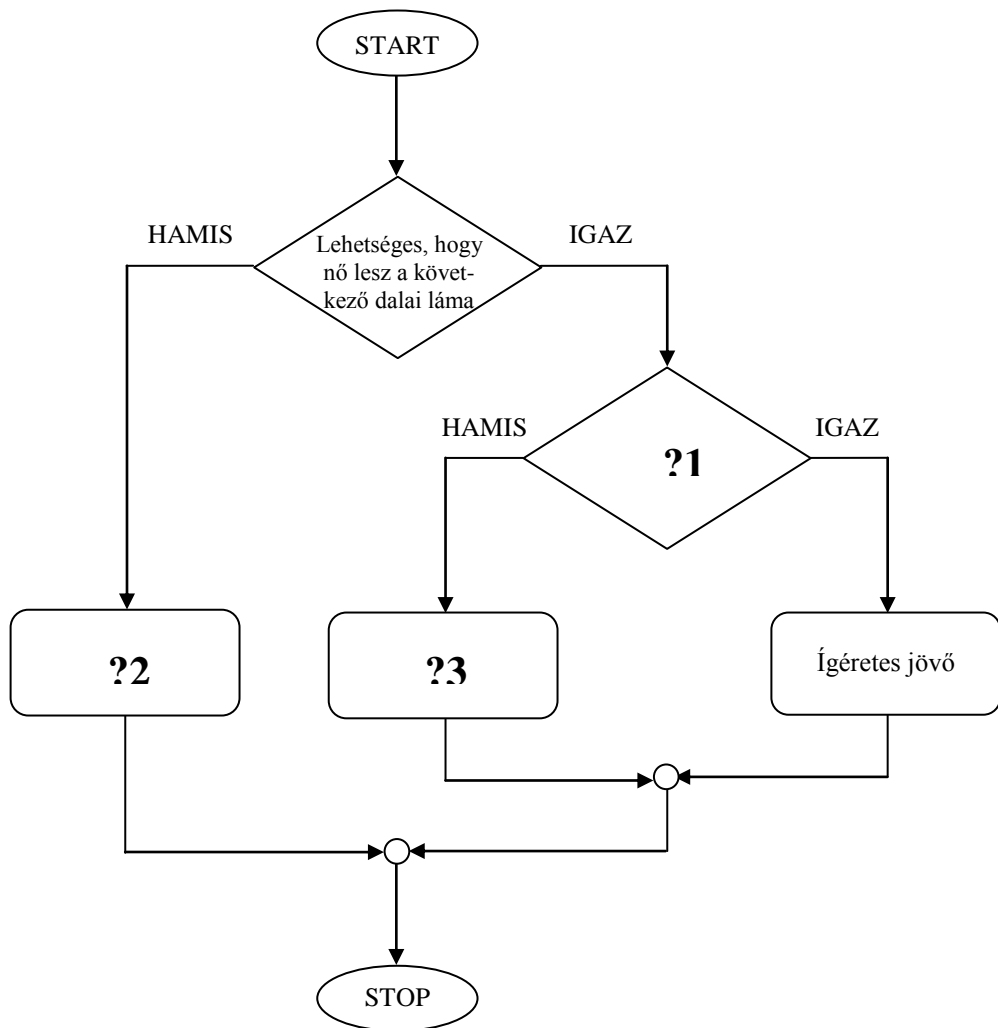
1. Ki dolgozta ki a számítógép működési elveit?
2. A számítógép működésének megismeréséhez milyen ismeretekre van szükségünk?
3. Az összeadás művelete milyen szerepet tölt be a számítógép működésében?
4. Melyek a számítógép legfőbb részei?
5. Értelmezd: „a feldolgozóegység a számítógép lelkének tekinthető” megállapítást!
6. A szöveg alapján magyarázd meg a memória szó jelentését!
7. Mi a processzor szerepe a számítógép működésében?
8. A számítógép milyen egységei szolgálják az adatok tárolását?
9. A szöveg alapján értelmezd a perifériák szó jelentését!
10. A köznyelvben milyen jelentésben használod a periféria szót? Írj rá példát!
11. Milyen egységek teremtik meg a kapcsolatot a számítógép és a felhasználók között?
12. Melyek a számítógép központi egységéhez kívülről csatlakozó eszközök?
13. Milyen szempont alapján csoportosítják a háttértárolókat?
14. Határozd meg a szöveg típusát!
15. Mi jellemzi a szöveg nyelvi megformálását (stílusát)? Sorolj fel három jellemzőt!
16. A mellékelt ábra segíti-e a szöveg megértését? Indokold néhány mondatban álláspontodat!
17. Az ábrán milyen szerepük van a nyilaknak?



**2/a. FELADAT**

*Lehet-e nő a következő dalai láma? – kérdezte a BBC a mostani Dalai lámától. Igen, nyugtatott meg az elűzött tibeti főpap, de akármilyen nő azért nem lehetne. Csakis egy szép arcú nő. A 14. Dalai láma, Tendzin Gyaco szerint ugyanis nehéz világban élünk, ahol a nőknek egyre fontosabb szerepet kell játszaniuk, de azért fontos, hogy szép legyen az arcuk. „Ha egy női dalai láma jönne, akkor neki nagyon szépnek kell majd lennie, különben nem lenne sok haszna” – fűzte tovább gondolatait a vallási vezető. A BBC riportere vissza is kérdezett, hogy viccel-e, de nem.*

**2/a. Feladatlap:** Töltsd ki a kérdőjellel megjelölt helyeket az alapján, hogy milyen jóslatot tett – a sorok között– Tendzin Gyacoa világ jövőjét illetően.



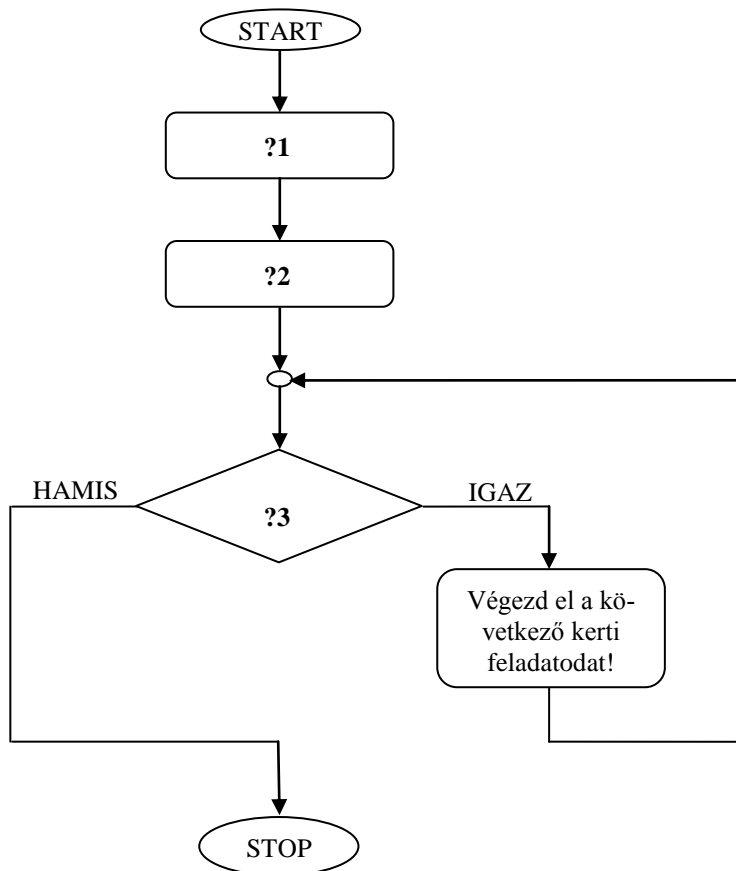
**2/b. FELADAT**

*A szegény ember és szegény asszony alig hagyták el a nagynéni házát, Panka mindjárt be lett fogva dolgozni. Ki kellett takarítani a hatalmas házat, reggelit készíteni a nagynéninek, ha ezzel végezett a kertben várta a munka. „Nem szeretném, ha csak egy pillanatra is abbahagynád a munkát. Tudd meg, nálam nem lehet lazáslni. Ezenkívül jól jegyezd meg, nem szeretnék meghallani nevetni, mert a nevetés tilos a házamban. Éppen indulok a bíróhoz, hogy tiltsa meg a nevetést, kacarászást a házam előtt. Amíg távol leszek, végezd a dolgodat, mert ha nem, nagyon megjáród!*

*A kislány szorgalmasan nekiállt a kirótt munkának, de nagynénje szigora sem tudta elvenni a jókedvét. A kertben vidáman énekelt és nevetgél. Minden mosolyra készítette. A madarak dala oly szép volt, hogy Panka valósággal elolvadt tőle. Feladataival gyorsan végzett. Unatkozott, nem tudta mivel üsse el az időt. Panka sétálgatni kezdett a kertben az almafák alatt. Megcsodált mindent, ami az útjába került. Beszélgetett a fákkal, kedvesen megcírógatta kerges törzsüket. Ha egy kismadár szállt valamelyik almafára, ámulva hallgatta énekét, és ő is füttyölt neki egy szép népdalt.*

**2/b. Feladatlap:** A mellékelt ábra azt mutatja, hogy milyen munkamenetet szánt a nagynéni –a szóban forgó napra– az újdonsült „vendégének”. Töltsd ki a kérdőjellel megjelölt helyeket.

**(4)** Bár Panka szófogadó, aki minden feladatát lelkiismeretesen elvégzi, milyen eshetőségre nem számított a nagynéni, amiért –várakozásaival ellentétben– nem fogja munkavégzés közben találni a kislányt?



### 3. FELADAT

Adott két természetes szám,  $A$  és  $B$  ( $A < B$ ). Az  $A$  számot minden lépésben felezzük, a  $B$ -t pedig duplázzuk (a felezésnél a maradékot, ha van, elhagyjuk). Mindezt addig ismétljük, amíg az  $A$  szám 1 nem lesz. Adjuk össze a „duplázási számsor” ( $B, 2B, 4B, \dots$ ) azon elemeit, amelyeknél a megfelelő pozíciójú elem a „felezési számsorban” ( $A, A/2, A/2/2, \dots$ ) páratlan volt.

Milyen eredményhez jutunk az  $A=26$  és  $B=34$  esetben? Milyen eredményhez jutnánk, ha a nagyobbik számot feleznénk, és a kisebbiket dupláznánk? Hogyan értelmeznénk az eredményt?