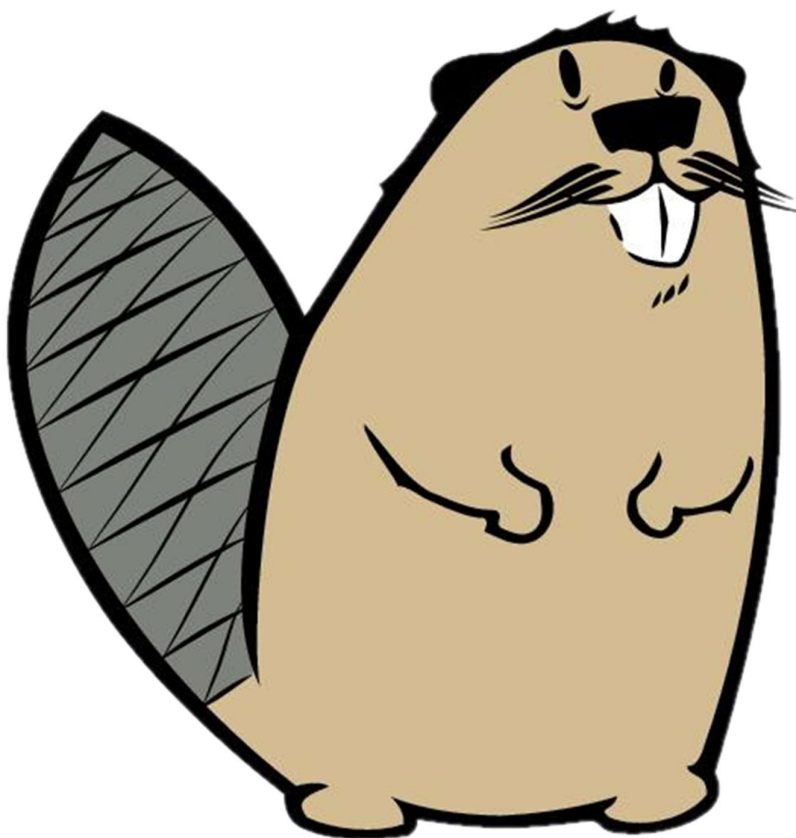


# **HÓDÍTSD MEG A BITEKET!**

**INFORMATIKAI GONDOLKODÁST TÁMOGATÓ, NEMZETKÖZI BEBRAS  
KEZDEMÉNYEZÉS MAGYAR MEGVALÓSULÁSA**



**2018**



### MI IS AZ E-HÓD?

Az e-HÓD/HÓDítsd meg a biteket a nemzetközi BEBRAS-kezdeményezés magyar partnere.

A nemzetközi Bebras, melyhez 2017-ben már 50 ország kapcsolódott, 2015-ben elnyerte az Informatics Europe „Best Practices in Education” díját.

A kezdeményezés alapja Dr. Valentina Dagiene litván professzor által életre keltett verseny, melynek célja, hogy rövid, gyorsan (kb. 3 perc alatt) megérthető és megoldható feladatokkal megvalósítsa az alábbiakat:

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

A kérdések három nehézségi szinten csak strukturált és logikus gondolkodást igényelnek, semmilyen különleges informatikai tudás nem szükséges a megválaszoláshoz. A feladatok érdekes problémákat mutatnak be. Nem tesztek, inkább szórakoztató gondolkodtató feladványok.

Magyarországon 2018-ban nyolcadik alkalommal, öt korcsoportban vehettek részt a diákok 4-től 12. osztályig.

A versenyt az ELTE IK T@T Labor és az NJSZT Közoktatási Szakosztálya szervezi.

Az alábbi dokumentumban a 2018-as magyar verseny feladatai és megoldásai találhatóak.

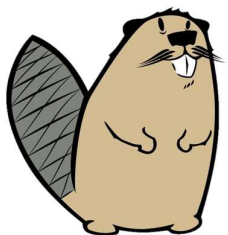
További információkért látogasson el a [HTTP://E-HOD.ELTE.HU/](http://E-HOD.ELTE.HU/) weboldalra, vagy írjon email-t az [info@e-hod.elte.hu](mailto:info@e-hod.elte.hu) címre.

### RÉSZVÉTEL

A részvétel mindenki számára ingyenes.

A verseny november második hetében kerül lebonyolításra, osztályonként kiválasztható, hogy az adott héten melyik napon mikor oldják meg a feladatokat (8:00 és 14:00 között). Ezzel biztosítható, hogy akár egy tanóra keretein belül tudjanak részt venni egész osztályok.

A résztvevő diákoknak egy-egy internet kapcsolattal rendelkező számítógépre van szükségük. A feladatok megjelenítése és elküldése minden böngészőn működik. A verseny befejezése után, a hód hetet követően kerülnek nyilvánosságra a megoldások, melyek lehetőség szerint átbeszélhetők ugyancsak akár egy tanóra keretein belül.





## SZABÁLYOK

- A verseny lebonyolítása iskolai helyszíneken történik.
- A résztvevők online kapják meg és válaszolják meg a kérdéseket;
- A versenyre fordítandó idő 45 perc, 18 feladat három nehézségi szinten: könnyű, közepes és nehéz (legkisebb korosztályban 11 feladat);
- A verseny alatt semmilyen más számítógépes program, alkalmazás nem használható;
- A verseny során nyugalmas környezetet kell biztosítani;
- A terem a verseny során nem hagyható el;
- Az esetleges számítógéppel, internettel kapcsolatos észrevételeket a kontakt személynek kell összegyűjtenie és továbbítani a szervezők felé;
- A verseny célja: minél több pont összegyűjtése helyes válaszok megjelölésével, helytelen válaszok esetén pontlevonás történik;
- A kérdések tetszőleges sorrendben megválaszolhatók;
- A kérdések, problémák megértése a feladat részét képezi. Ezért a feladatok megbeszélése és értelmezéssel kapcsolatos kérdések nem megengedettek;
- A megoldások a verseny befejezése után, a hód hetet követően kerülnek nyilvánosságra.

## ÉRTÉKELÉS, PONTOZÁS

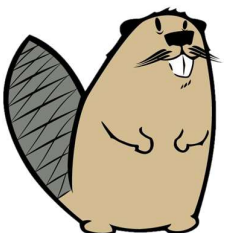
A Kishód korcsoportban 11, minden más korcsoportban 18 feladatot kell megoldani három nehézségi szinten. Minden helyes válasz pontot ér, minden helytelen válaszáért pontlevonás jár.

Nem megválaszolt kérdés esetében az összpontszám változatlan marad.

Az alábbi táblázat mutatja, hogy a feladatok nehézségétől függően hány pont kerül jóváírásra, illetve levonásra:

	Könnyű	Közepes	Nehéz
Helyes válasz	6 pont	9 pont	12 pont
Helytelen válasz	-2 pont	-3 pont	-4 pont

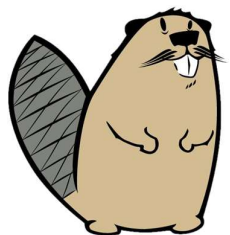
Összesen (18 feladat esetében) maximum 162 pont érhető el.





**TARTALOMJEGYZÉK**

Mi is az E-HÓD?.....	2
Részvétel.....	2
Szabályok .....	3
Értékelés, Pontozás .....	3
Tartalomjegyzék.....	4
Feladatok.....	5
Hód a mérlegen (2011-IL-01) .....	7
Sütemény elvitelre (2013-AT-11) .....	9
Evezős verseny (2013-JP-04) .....	11
LISA-k feltöltése (2014-DE-08).....	13
Kalózvadászat (2015-SI-07) .....	15
Át a sötét barlangon (2016-CH-04a) .....	17
Pontgyűjtés (2017-CA-12) .....	19
Mosolyt kérek! (2017-DE-02).....	21
Kártyafordító (2018-BE-01a).....	23
Sorok és oszlopok (2018-BE-03).....	25
Próbaterv (2018-be-04) .....	27
Fák egy körben (2018-CA-02) .....	29
Simon mondja (2018-CA-03a).....	31
Szomszédok (2013-AT-11).....	33
Társasjáték (2018-CA-06) .....	35
Számítógépes játék (2018-CH-03).....	37
Barátlátogatás (2018-CH-05) .....	39
Felvonó a Matterhorn-on (2018-CH-07) .....	41
Klára virágai (2018-CH-09) .....	43
Ruhakupac (2018-CH-10b) .....	45
Könyv csere-bere (2018-CN-02).....	47

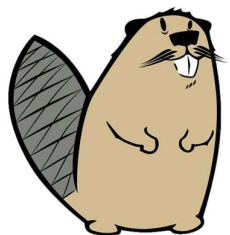




A lehető leghosszabb szólánc (2018-CZ-08c) .....	49
Z bolygó (2018-DE-02) .....	51
Az elveszett autó (2018-DE-03) .....	53
Jelszó biztonság (2018-DE-04) .....	55
A hazúgot hamar utoléri (2018-HR-05) .....	57
Ada ceruzái (2018-HU-01) .....	59
Vízesés (2018-HU-03) .....	61
Quattris (2018-HU-04) .....	63
Hasonló ételek (2018-HU-05) .....	65
Biztonsági zár (2018-HU-06) .....	67
Lámpák és kapcsolók (2018-HU-08) .....	69
Végtelen fagyalttölcsér (2018-IE-01) .....	71
Szigorúan titkos (2018-IT-04) .....	73
Hód kapitány napja (2018-KR-08) .....	75
Hód tó (2018-LT-02) .....	77
Két hód dolgozik (2018-LT-04) .....	79
Soundex (2018-PK-06) .....	81
29-es házikó (2018-SK-07) .....	83
Üzenet a Suli Menedzsment RendSzerből (2018-TR-02) .....	85
A földönkívüli (2018-TR-06) .....	87
Úthálózat (2013-TW-02) .....	89
Kifestő (2018-UK-01) .....	91
Napirend (2018-UK-02) .....	94
Három barát (2018-VN-03) .....	96
Találd meg a repedést (2018-ZA-01) .....	98
Támogatóink, köszönetnyilvánítás .....	100

## FELADATOK

**Kishód:** 2011-IL-01, 2018-CA-03a, 2018-CH-10b





2018-CA-02, 2018-HU-01, 2018-HU-05, 2018-UK-01

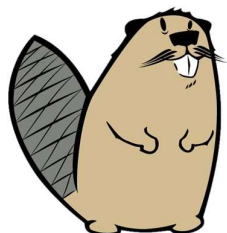
2018-CH-09, 2018-HU-06, 2018-TR-02, 2018-TW-02

**Benjámín:** 2011-IL-01, 2018-CA-02, 2018-HU-05, 2018-HU-06, 2018-TR-06, 2018-UK-01  
2018-CH-09, 2018-DE-02, 2018-DE-04, 2018-IE-01, 2018-TR-02, 2018-TW-02  
2013-AT-11, 2013-JP-04, 2018-CH-07, 2018-HR-05, 2018-HU-03, 2018-LT-02

**Kadét:** 2017-DE-02, 2018-DE-02, 2018-DE-04, 2018-HU-01, 2018-TW-02, 2018-ZA-01  
2013-AT-11, 2018-CH-07, 2018-HR-05, 2018-HU-03, 2018-LT-02, 2018-TR-02  
2016-CH-04a, 2018-CA-05, 2018-CH-05, 2018-HU-08, 2018-KR-08, 2018-SK-07

**Junior:** 2014-DE-08, 2018-HU-03, 2018-KR-08, 2018-LT-02, 2018-TR-02, 2018-VN-03  
2016-CH-04a, 2018-CA-05, 2018-CA-06, 2018-CH-05, 2018-HU-08, 2018-LT-04  
2018-BE-03, 2018-CN-02, 2018-CZ-08c, 2018-HU-04, 2018-PK-06, 2018-UK-02

**Senior:** 2018-CA-06, 2018-CH-05, 2018-CH-03, 2018-HU-08, 2018-LT-04, 2018-TR-02  
2018-BE-01a, 2018-CN-02, 2018-CZ-08c, 2018-DE-03, 2018-HU-04, 2018-UK-02  
2015-SI-07, 2017-CA-12, 2018-BE-03, 2018-BE-04, 2018-IT-04, 2018-PK-06

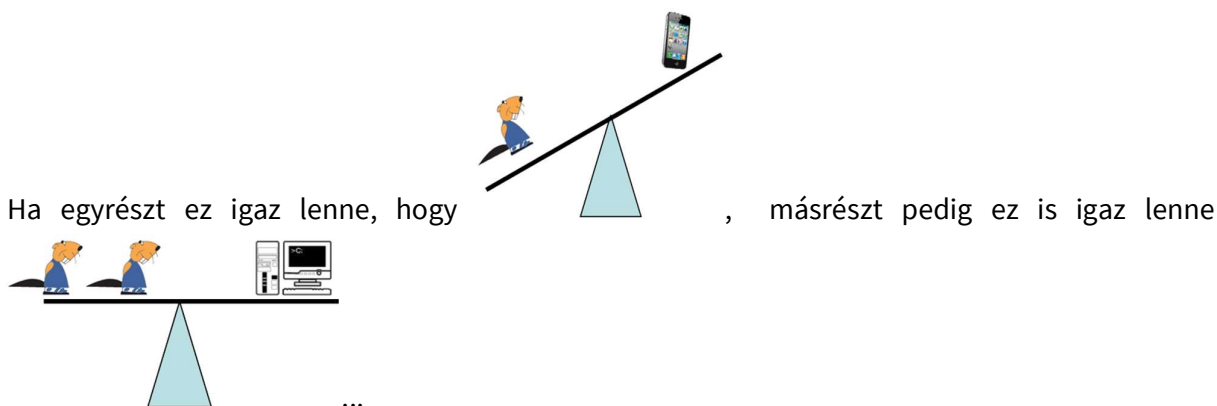
















### HÓD A MÉRLEGEN (2011-IL-01)

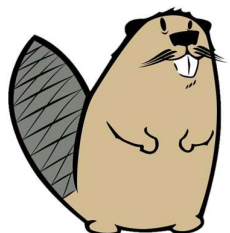
KISHÓD – KÖNNYŰ

BENJAMIN – KÖNNYŰ







... akkor **minek kellene még igaznak lennie?**

- A.  nehezebb, mint  és nehezebb, mint 
- B.  nehezebb, mint  és könnyebb, mint 
- C.  könnyebb, mint  és nehezebb, mint 
- D.  könnyebb, mint  és könnyebb, mint 





### „C” válasz a helyes

Az első kép azt mutatja: a hód (  ) nehezebb, mint a mobiltelefon (  ). A második kép pedig azt mutatja: számítógép (  ) körülbelül kétszer olyan nehéz, mint a hód (  )

### MIÉRT INFORMATIKA?

Ha ez és amaz így volna, valamint még ez is meg az is... akkor milyen végkövetkeztetések lennének lehetségesek? A logikai következtetés az informatikai rutinmunka része. Egyszerűbb esetekben a fejünket használjuk.

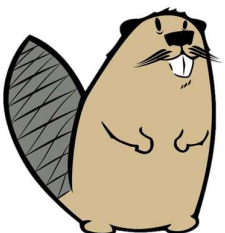
Igen sok feltételezésnél, valamint a nem klasszikus logikában (például a fuzzy logikában, a nem monoton logikában, a többértékű logikában) a számítógép nagyszerű eszköznek bizonyul ahhoz, hogy megmaradjon a rálátásunk a dolgokra. A következtetésekhez vezető feltételezéseket azonban az ember választja ki. Ha ezek hamisak, ellentmondásosak vagy nem teljesek, a számítógép értelmetlen következtetéseket von le. Az informatikusok ezt GIGO-elv néven ismerik: ha szemetet adunk be, szemetet kapunk: garbage in, garbage out.

### KULCSSZAVAK

Logika, gépi döntés

### WEBOLDALAK

[https://hu.wikipedia.org/wiki/Matematikai\\_logika](https://hu.wikipedia.org/wiki/Matematikai_logika)





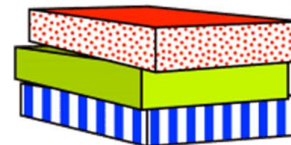
### SÜTEMÉNY ELVITELRE (2013-AT-11)

BENJAMIN – NEHÉZ

KADÉT – KÖZEPES

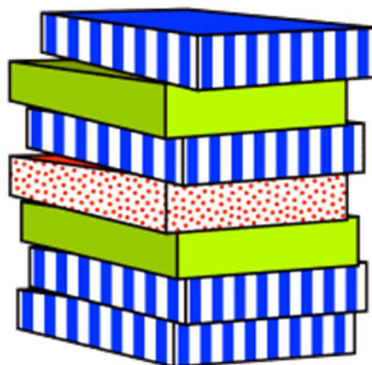
Tibi cukrász, és egyszerre mindig három süteményt süt.

Amint kész van a három sütemény, Tibi színes dobozokba csomagolja őket. Ezeket rögvest egymásra rakja. Ezt látod a képen is.

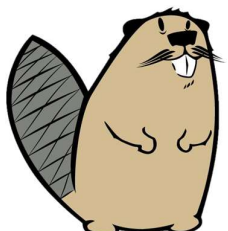


A hármas kupacot azonnal Tomihoz, az eladóhoz viszi. Tomi a hármas kupacot az eladásra szánt dobozok tetejére teszi. Ha Tomi elad egy süteményt, mindig a legfelső csomagot veszi le az eladásra szánt dobozok közül.

**Legkevesebb hány süteményt adott el Tomi, ha így néz ki az eladásra szánt csomagok kupaca?**



- A. 4 süteményt.
- B. 5 süteményt.
- C. 6 süteményt.
- D. 7 süteményt.





### „B” válasz a helyes

Tomi legalább 5 süteményt adott el. Ezek hiányoznak, ha az eladásra szánt csomagok kupacát összehasonlítjuk a teljes számú elkészült, eladásra odakészített csomagok kupacával.

A teljes számú azt jelenti, hogy egyetlen hármassal sem hiányzik egy sütemény sem. Az is lehet, hogy Tomi nyolc vagy tizenegy süteményt is eladott már, hiszen ha egy hármassal kupacból mindhárom süteményt eladta, azt nem tudjuk leolvasni az eladásra szánt csomagok kupacából.

### MIÉRT INFORMATIKA?

A verem kifejezést a tárolás egyik formájára használjuk, ahol az utoljára beolvasott elem, mint első, újra elolvasásra kerül. Ezt LIFO-nak nevezzük: Last In First Out.

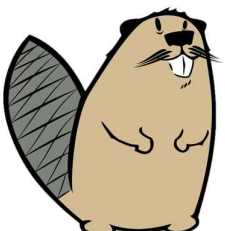
A verem koncepciót az informatika gyakran alkalmazza, például alprogramok behívásánál. A számítógép egy veremben jelöli, hol kell a programnak továbbfutnia, ha az alprogram elkészült. A verem tehát praktikus, mert így az alprogram még egy másik al-alprogramot is képes behívni és így tovább...

### KULCSSZAVAK

Verem, Adatszerkezetek

### WEBOLDALAK

[http://hu.wikipedia.org/wiki/Verem\\_%28adatszerkezet%29](http://hu.wikipedia.org/wiki/Verem_%28adatszerkezet%29)





### EVEZŐS VERSENY (2013-JP-04)

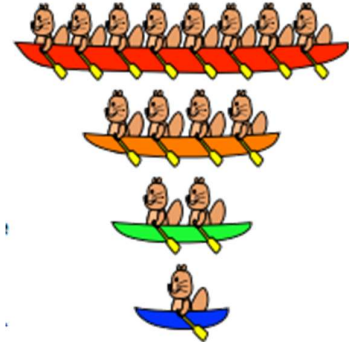
#### BENJAMIN – NEHÉZ

Néhány hód egy evezősversenyen szeretne elindulni.

Összesen négy hajójuk van, minden hajóosztályban egy: egy 8 személyes, egy 4 személyes, egy 2 személyes és egy 1 személyes.

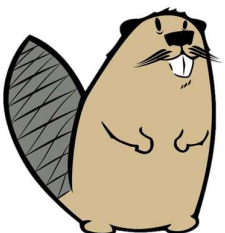
A verseny szabályai előírják, hogy minden hód csak egy hajóosztályban indulhat.

A hódok edzőjének fel kell írnia, hogy egy adott hajóosztályban elindulnak (1) vagy nem indulnak el (0) a versenyzői. A legnagyobb hajóval kezd, aztán a második legnagyobb következik, és így tovább. Ha például 10 hód szeretne elindulni a versenyen, ezt írná fel: 1010



**Ezúttal 13 hód szeretne elindulni a versenyen. Mit írjon fel az edző?**

- A. 0111
- B. 1011
- C. 1101
- D. 1110





**„C” válasz a helyes**

$8+4+0+1=13$  hód

„A” válasz lenne a helyes, ha 7 hód indulna ( $0+4+2+1=7$ ).

„B” válasz lenne a helyes, ha 11 hód indulna ( $8+0+2+1=11$ ).

„D” válasz lenne a helyes, ha 14 hód indulna ( $8+4+2+0=14$ ).

**MIÉRT INFORMATIKA?**

A kettes számrendszer a megszokott tízes számrendszerhez hasonlóan egy helyiértéken alapuló számrendszer, de a tíz lehetséges számjegyből (0-9) csupán kettőt (0 és 1) használ. Az n-edik hely helyiértéke tehát nem  $10^n$ , hanem  $2^n$ . Hogy egy számot kettes számrendszerből tízes számrendszerbe átírjunk, egyszerűen meg kell szorozni minden számjegyet a neki megfelelő helyiértékkel, így:

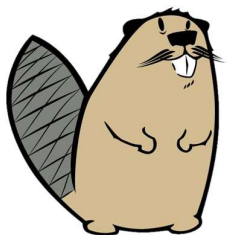
$$1101 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2 = 8 + 4 + 0 + 1 = 13.$$

**KULCSSZAVAK**

Kettes számrendszer, Kódolás

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Kettes\\_számszrendszer](https://hu.wikipedia.org/wiki/Kettes_számszrendszer)



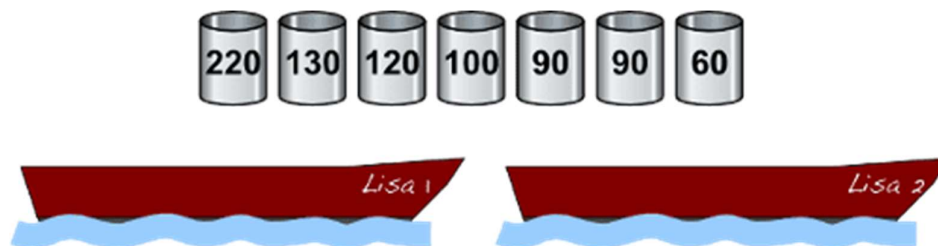


LISA-K FELTÖLTÉSE (2014-DE-08)

## JUNIOR – KÖNNYŰ

Bertalané és Barnabásé, a két halászné a "Lisa1" és "Lisa2" hajó – a két Lisa.

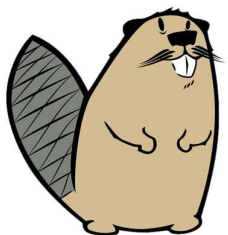
Mindkét hajó legfeljebb 300kg-ig terhelhető.



Bertalané és Barnabásé a két hajóval el kell szállítania pár hordó halat. A szállítást súly alapján fizetik.

**Maximum mennyi halat tudnak egyszerre a két hajóval elszállítani?**

- A. 810kg
- B. 600kg
- C. 590kg
- D. 530kg





**„C” válasz a helyes**

Összesen 590 kg halat tudnak elszállítani:  $120+90+90=300$  az egyik hajón és  $130+100+60=290$  kg a másik hajón.

Vigyázz, ne legyél mohó! Ha legelőször a legnehezebb hordókat veszed a hajók megtöltéséhez, maximum  $220+60=280$  és  $130+120=250$ , azaz összesen 530 kg halat tudsz felpakolni.

590 kg-nál több hal nem pakolható fel. Ehhez mindkét hajónak 300 kg-ot kellene szállítania. Nincs azonban csak egy lehetőség ( $120+90+90$ ) az adott hordókból 300 kg-nyi halat összerakni.

**MIÉRT INFORMATIKA?**

Sok ember el van ragadtatva attól, hogy dolgokat optimalizáljon – gyakran költséget spóroljon és a bevételét maximálja. A nem egyszerű problémák esetében legtöbbször számítógépes programokat alkalmaznak az optimalizáláshoz: a legrövidebb út, az optimális terhelés, az ideális órarend megtalálásához. Egyes optimalizálási problémák úgynevezett „mohó” algoritmussal oldhatóak meg. Ennél a megoldáshoz vezető minden lépés (itt a hordók kiválasztása) úgy kerül meghatározásra, hogy annyi profitot hozzon, amennyi lehetséges – ettől mohó.

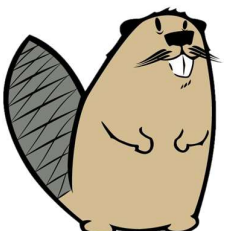
Ami szép az informatikában: a legtöbb esetben a mohóság nem segít többet és összetettebb algoritmusok alkalmazása szükséges az optimális megoldás megtalálásához. Egyes problémák esetében bizonyítható, hogy az olyan algoritmusok, melyek az optimális megoldást megtalálják, magától a számítógéptől elfogadhatatlanul nagy erőfeszítést igényelnek. Sok ilyen nehéz optimalizálási problémára az informatika hatékony algoritmusokat alakított ki, melyek korántsem a legjobb (optimális), de kimutathatóan nagyon jó, a majdnem optimális megoldást találják meg.

**KULCSSZAVAK**

Optimalizálás

**WEBOLDALAK**



<https://hu.wikipedia.org/...>





### KALÓZVADÁSZAT (2015-SI-07)

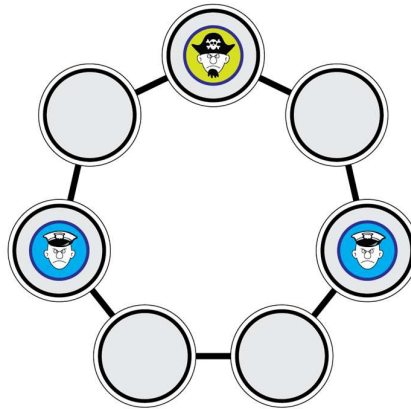
SENIOR – NEHÉZ

A Kalózvadászat játék a következőképpen folyik: a “rendőrség”  és a “kalóz”  felváltva lépnek.

Amikor a rendőrség következik, a rendőrök egyikének a mellette lévő szabad helyre kell lépnie. A kalóz mindig két mezőt lép tovább.

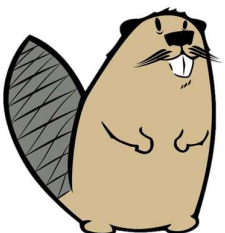
A játék akkor ér véget, ha a kalóz már csak olyan mezőre tud lépni, ahol rendőr áll. Ek a kalóz vesztett, a rendőrség pedig nyert. A rendőrség tehát megpróbálja a kalózt ilyen helyzetbe kényszeríteni.

A játék a képen látható felállással kezdődik és a rendőrség kezd.



Tegyük fel, hogy a kalóz nem hibázik. **Van esélye a rendőrségnek, hogy nyerjen?**

- A. A rendőrség 2 lépésben nyerhet.
- B. A rendőrség 3 lépésben nyerhet.
- C. A rendőrség 5 lépésben nyerhet.
- D. A rendőrségnek nincs esélye.

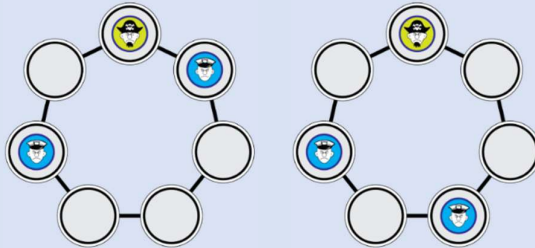




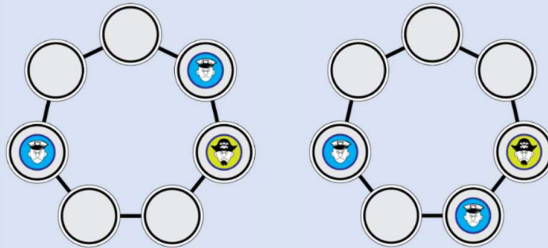
**„D” válasz a helyes**

Tegyük fel, hogy a rendőrségnek van esélye.

A játék a bemutatott állapotban van és a kalóz következik. Milyen lépéssel kényszerítheti a rendőrség a kalózt nyerő helyzetbe? Az egyik rendőrnek egy mezőt kell mozognia felfelé vagy lefelé. Mivel a játéktér szimmetrikus, mindegy, melyiknek. A lépés előtt tehát a játék a következő állapotok egyikében lehetett:



Mit léphetett ezelőtt a kalóz, hogy ilyen helyzetbe került? Mindkét esetben csak jobbról jöhetett (balról a rendőr állt a mezőn). Tehát az alábbi állapotok valamelyikében volt a játék:



Csak az egyik ilyen helyzetből kerülhet a rendőrség nyerő helyzetbe. Mivel azonban a kalóz nem hibázik, ezekben a helyzetekben nem jobbra, hanem balra mozdulna, tehát nem hozná nyerő helyzetbe a rendőrséget.

**MIÉRT INFORMATIKA?**

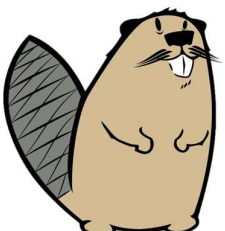
Sok kétjátékos játékot a számítógép ellen is játszhatunk. Az ezekre írt programok kiszámolják a lépéseiket, melyekkel az éppen aktuális helyzetből kiindulva nyerhetnek. A program értékeli saját lépéseit és felteszi, hogy az ellenfél nem hibázik (mint a feladatban a kalóz). Ha ezek a játékok nagyon összetettek, (mint pl. a sakk), nem lehetséges minden lépést előre kiszámolni. Egyes kétjátékos játékok esetében a programok jobban játszanak, de van, ahol még az ember nyer.

**KULCSSZAVAK**

Játék, Algoritmus

**WEBOLDALAK**

<https://hu.wikipedia.org/wiki/Játékelmélet>  
[https://hu.wikipedia.org/wiki/Minimax\\_elv](https://hu.wikipedia.org/wiki/Minimax_elv)



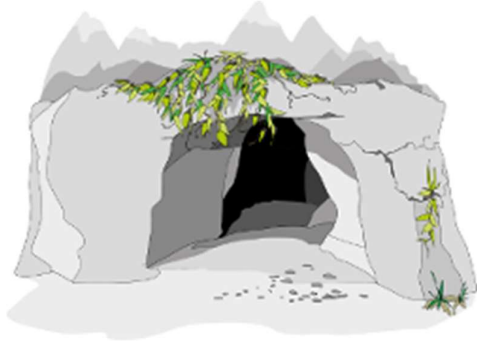


### ÁT A SÖTÉT BARLANGON (2016-CH-04A)

KADÉT – NEHÉZ

JUNIOR – KÖZEPES

Anna és Beni a szüleikkel túrázni mennek. Útközben egy barlangon kell keresztülmászniuk. Tapasztalatból tudják, hogy ez mindannyiuknak különböző időbe telik: Annának 10 perc, Beninek 5 perc, az anyukájuknak 20 perc és az apukájuknak 25 perc.

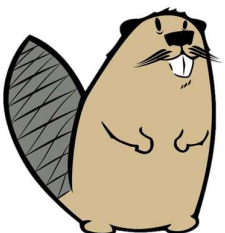


A keskeny és sötét barlangon biztonságosan csak egy vagy két hód mászhat át egy időben. Mivel csak egy zseblámpájuk van, több fordulót kell tenniük. Amikor ketten mennek, mindig annyi időre van szükségük, amennyire a lassabbnak.

Az egész család szeretne a barlang másik oldalán lenni egy órán belül, mielőtt az elemlámpa lemerül. Anna úgy gondolja, sikerülhet, méghozzá 5 átmenettel!

**Kik NEM mehetnek együtt, ha azt akarják, hogy Anna terve sikerüljön?**

- A. Beni és apukája.
- B. Apuka és anyuka.
- C. Anna és Beni.
- D. Mindegy hogy mennek, 1 óra alatt akkor sem sikerül.





### „A” válasz a helyes

Anna megoldotta az alagútproblémát egy frappáns ötlettel! Ez az ötlet meg kell feleljen bizonyos feltételeknek (az átjutóknál mindig kell legyen egy lámpa), és ennek a tervnek a végrehajtásához figyelembe kell venni a korlátozó tényezőket (lámpa működési ideje és az idő).

Csak úgy lehetséges, hogy mindannyian 60 perc (1 óra) alatt átérjenek, ha az anya és az apa (a két leglassabb) csak egyszer – és egyszerre – vannak a barlangban. Ez csak akkor megoldható, ha korábban Anna és Beni már átment és az egyikük hozza vissza a zseblámpát. Az, hogy melyikük hozza vissza a szülők előtt és melyikük utánuk a lámpát, az idő szempontjából mindegy (lásd a táblázatokat).

út	hód	hód	perc
1. oda	Anna	Beni	10
2. vissza	Beni		15
3. oda	Hódpapa	Hódmama	40
4. vissza	Anna		50
5. oda	Anna	Beni	60

út	hód	hód	perc
1. oda	Anna	Beni	10
2. vissza	Anna		20
3. oda	Hódpapa	Hódmama	45
4. vissza	Beni		50
5. oda	Anna	Beni	60

### MIÉRT INFORMATIKA?

Az informatikában egyre összetettebb rendszereket fejlesztenek, melyek saját működésüket maguktól (autonóm módon) meg kell, hogy tervezzék. Mindezt adott feltételek mellett, rendelkezésre álló eszközök segítségével. Ezek a rendszerek jóval szélesebb körűek, mint ez a feladat. Például az önjáró autó: meg kell terveznie az útvonalat, hogy célba érjen, szem előtt tartva, hogy mindig legyen elég energiája.

A független és biztos tervezésnek etikailag és jogilag is elfogadhatónak kell lennie, ha az autonóm rendszer működése emberekre és más élőlényekre is kihat.

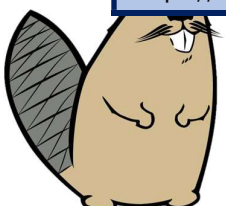
Hasonló elgondolások vezették az író, Isaac Asimovot 1942-ben, az első számítógépek készítésekor. Ő vezette be, többek között ezt a törvényt: „A robotnak nem szabad kárt okoznia emberi lényben vagy tétlenül tőnnie, hogy emberi lény bármilyen kárt szenvedjen.” Mivel az önjáró autó is egy robot, szintén be kell tartania az előbb említett szabályt. Az önvezető autó navigációját egyébként számos érzékelő és modern navigációs eszköz, így például radar, lézerradar, GPS segítségével oldják meg.

### KULCSSZAVAK

Logika, gépi döntés

### WEBOLDALAK

[https://hu.wikipedia.org/wiki/Matematikai\\_logika](https://hu.wikipedia.org/wiki/Matematikai_logika)





**PONTGYŰJTÉS (2017-CA-12)****SENIOR – NEHÉZ**

Egy népszerű gondolkodós játék leírása következik.

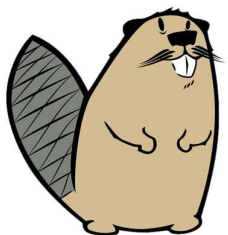
Adott egy táblázat (lásd lentebb), amelyben az S (start) mezőről kell eljutni a C (cél) mezőre úgy, hogy minden lépésben csak jobbra vagy felfele léphetsz, ahogy a nyilak is mutatják.

Minden cellában van egy szám, ami az adott mezőn megszerezhető pontszámot jelzi. Válaszd úgy ki a célba vezető utat, hogy a lehető legtöbb pontot gyűjtsd össze!

	2	0	1	1	<b>C</b>
	1	2	0	2	3
	2	2	0	2	1
	3	1	0	2	0
↑	<b>S</b>	0	1	3	0
					→

**Maximálisan hány pont érhető el?**

- A. 10
- B. 12
- C. 14
- D. 16





**„C” válasz a helyes**

Legfeljebb 14 pont gyűjthető össze. A következő táblázat az adott mezőig maximálisan megszerezhető pontszámot tartalmazza, a jobb felső sarokban pedig a végső, helyes megoldás látható. Ezt a táblázatot akkor kapod meg, ha minden egyes mezőnél megvizsgálod a bal és alsó szomszéd maximumát, melyhez hozzáadod az eredeti táblázat mezőértékét. A helyes megoldást szürkével jelöltük:

	8	9	10	12	14
	6	9	9	11	14
	5	7	7	9	10
	3	4	4	6	6
	0	0	1	4	4

**MIÉRT INFORMATIKA?**

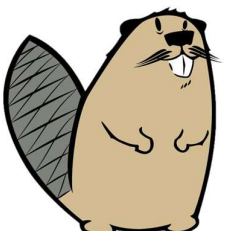
A leírt módszer nem feltétlen a leggyorsabb megoldási mód. Az agyunk nagyon hatékony a megoldást már-már magától adó feladatok megoldásában, ugyanis az eredmény néha azonnal látható. Ez a módszer azonban nem hibamentes, és csak az átlátható méretű táblázatok esetén működik. Ezzel szemben a fent leírt folyamat mindig helyesen működik, amennyiben az egyes lépéseket helyesen írták le. A program ezeket a lépéseket tetszőleges méretű táblázaton (informatikában mátrixon) végre tudja hajtani. Ez egy jó program ismértve.

**KULCSSZAVAK**

Algoritmikus gondolkodás

**WEBOLDALAK**

<https://hu.wikipedia.org/wiki/Algoritmus>





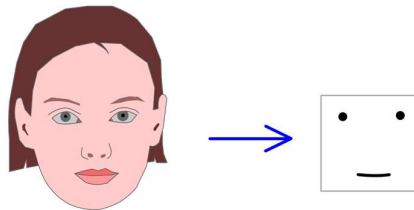
## MOSOLYT KÉREK! (2017-DE-02)

KISHÓD – NEHÉZ

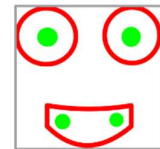
BENJAMIN – KÖZEPES

KADÉT – KÖNNYŰ

A hódok kifejlesztettek egy alkalmazást, amely felismer egy mosolygó arcot. Az alkalmazás először egy modellt készít az arképről, mely hasonlít egy szmájlira. Két pontból és egy vonalból áll, amelyek a szemek és a száj pozícióját ábrázolják.

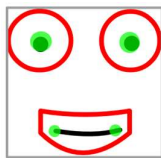


Ezt követően az alkalmazás egy zöld pontokból és piros vonalakkból álló sablon segítségével megvizsgálja a modellt.

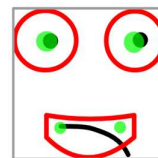


Mosolygós arként azonosítja, ha annak pontjai és vonala:

1. minden zöld pontot érintenek, és
2. nem metszenek egy piros vonalat sem.



Mosolygós arc

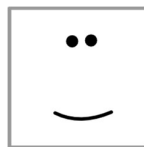


Nem mosolygó arc  
A száj metszi a piros vonalat.

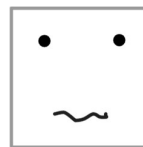
Az alkalmazás 4 képről készített modellt. Ezekből egyet azonosított mosolygós arként. Melyiket?



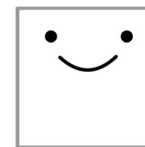
A)



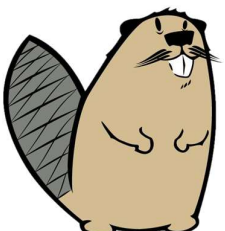
B)



C)



D)





**„C” válasz a helyes:**

Itt a szemek jó helyen vannak és a száj is a megadott keretben van. Az már más kérdés, hogy az arc láthatóan nem mosolyt ábrázol.

Az A kép esetében a száj kilóg a megadott keretből.

A B kép esetében a szemek távolsága nem megfelelő.

A D kép esetében a száj nincs a kívánt tartományban.

**MIÉRT INFORMATIKA?**

Az informatikusok olyan rendszereket fejlesztenek, amelyek kameraképeken arcokat, tárgyakat ismernek fel. Ez a technika igen nagy hatalommal bír, az azonosítást ma már mesterséges intelligencia végzi. Azonban az egyszerűbb szabályok - mint például ennek a feladatnak a kikötései - is nagyon fontosak.

A feladat a képfelismerő rendszerek nehézségeit mutatja be. Jelen esetben a szabályok túl egyszerűek ahhoz, hogy minden bizonyossággal mosolygós arcok kerüljenek kiválasztásra. A rendszer által mosolygósnak értelmezett két arc közül az egyik inkább mogorva arcot ábrázol. Ehhez hasonló hibákat a való életben használt arcfelismerő rendszerek is elkövethetnek például egy automatikus útlevél-ellenőrzéskor vagy egy hatósági eljárás során. 2016-ban egy ázsiai diák útlevélkérelmét utasították el, mert a számítógép nem ismerte fel a szemét az igazolványképen.

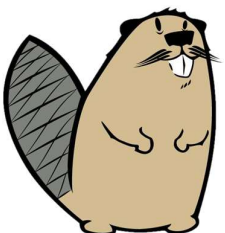
Forrás: <http://www.telegraph.co.uk/technology/2016/12/07/robot-passport-checker-rejects-asian-mans-photo-having-eyes/>

**KULCSSZAVAK**

Képfelismerés, algoritmus alkalmazása, gépi látás

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Gépi\\_látás](https://hu.wikipedia.org/wiki/Gépi_látás)





## KÁRTYAFORDÍTÓ (2018-BE-01A)

### SENIOR – KÖZEPES

Kaptál ajándékba egy pakli csupa egyforma lapból álló kártyát.

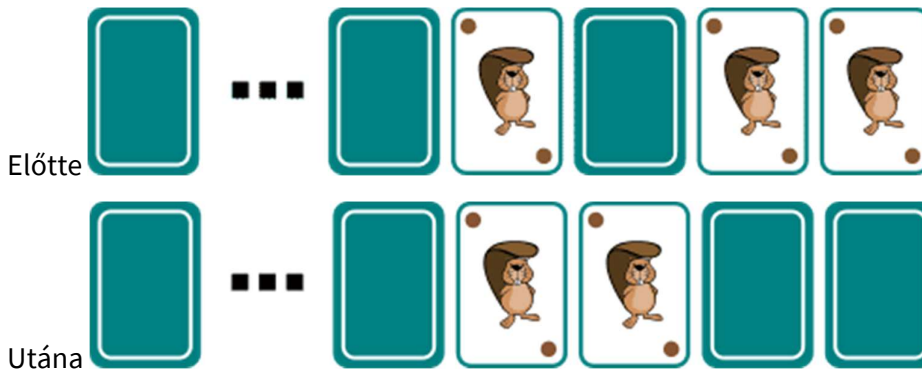


A kártyalapok így néznek ki: felfordítva: és lefordítva:

Ezekkel a kártyákkal „fordító”-t tudsz játszani. A kártyák egy sorban fekszenek előtted. Egy játékkörben a következőképpen mész végig a kártyákon:

- Ha az aktuális kártya le van fordítva, felfordítod. Ezzel a kör be is fejeződik, a többi kártya változatlanul marad.
- Ha az aktuális kártya fel van fordítva, fordítsd meg.

A képen láthatod, hogy változhatnak a kártyák egy játékkörben. Jobbról az első és a második kártya fel van fordítva, ezért lefordítod őket. A harmadik kártya lefele van fordítva, ezért felfordítod. Ezzel a lépésnek vége és a maradék kártyák változatlanul maradnak.

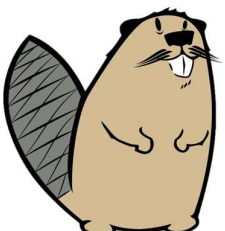


Ezúttal 16 lefordított kártyával kezdjük a játékot.

Hány kártyalap lesz 16 játékkör után felfele fordítva?



- egy sem.
- 1
- 8
- 16





**„B” válasz a helyes**

Mire emlékeztetnek ezek a kártyák? Egy sor ugyanolyan kártya, melyek vagy lefordítva vagy felfordítva lehetnek, a bináris (kettes számrendszerbeli) számot ábrázolnak. A bináris számok csak két számjegyből, a 0-ból és az 1-ből állnak: a lefordított kártya lehet a 0-s, a felfordított pedig az 1-es.

Hasonlóan a megszokott 10-es számrendszerhez minden helyiértéken egy bináris szám határozza meg, hogy azt a 2-es hatványértéket hozzá kell-e számolnunk a számhoz vagy sem. Például ha (jobbról) a harmadik helyen egy egyes áll, akkor bele kell számolnunk a harmadik kettőhatvány értékét: azaz a 22-t, mivel az első helyen a 20 áll.

A bináris számokat is ilyen módszerrel növelhetjük meg eggyel. A jobbról az első számjeggyel kezdjük:

- Ha az aktuális számjegy 0, akkor 1-é változtatjuk. Ezzel készen is vagyunk – a számunkat megnöveltük 1-gyel.
- Ha az aktuális számjegy 1, akkor 0-vá változtatjuk, és átlépünk a következő, balra elhelyezkedő számjegyre (átvitel).

Ez pontosan ugyanaz, mint a „fordító” játék egy köre. Egy játékkör tehát eggyel megnöveli a bináris számunk értékét, melyet a kártyákkal ábrázoltunk (reprezentáltunk).

A kezdetben az összes lefordított kártya az összes 0-ból álló számot, azaz a 0-t jelentette.

A kép mutatja az első négy játékkört, melyben tehát 1-től 4-ig számoltunk. Látható, hogy az 1, 2 és 4 számoknál (tehát a 2 hatványainál:  $2^0$ ,  $2^1$  és  $2^2$ ) pontosan egy kártya van felfordítva: tehát kettes számrendszerben egy olyan számnál, mely a 2 hatványa, csak azon a helyiértéken van egy 1-es, ami azt a hatványt jelenti.

16 játékkört követően azt a bináris számot akarjuk megjeleníteni, melynek az értéke a 16. Mivel  $16=2^4$ , ezért ez a szám jobbról az 5. helyen egy 1-est, egyébként pedig csupa 0-t tartalmaz: 0...010000. Tehát pontosan egy kártya lesz felfordítva.

**MIÉRT INFORMATIKA?**

Egy számítógép legkisebb memóriaegysége csak két állapotot vehet fel: BE vagy KI, IGAZ vagy HAMIS, 1 vagy 0. Minden adatot, ami egy számítógépben kerül tárolásra és feldolgozásra, bináris számjegyek sorozataként, azaz bináris számként ábrázolhatunk. Ezért van nagy jelentősége az informatikában a bináris számokkal való műveleteknek.

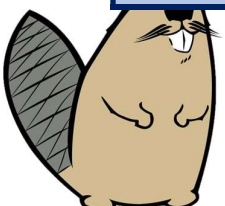
Mióta számítógépek vannak, az ilyen műveletek meglehetősen hatékonyan valósulnak meg. Léteznek olyan műveletek, melyekkel két bináris számot egymással összekapcsolunk, mint az összeadás, szorzás. De vannak olyanok is, melyekkel egy bináris számot megváltoztatunk, például amikor minden számjegyet egy hellyel balra tolunk (ez a 2-vel való szorzást adja eredményül) – vagy mint ebben a hód-feladatban a szám eggyel való növelése.

**KULCSSZAVAK**

Bináris számok, műveletek

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Kettes\\_számrendszer](https://hu.wikipedia.org/wiki/Kettes_számrendszer)

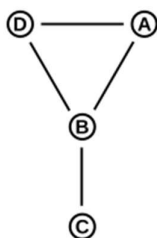
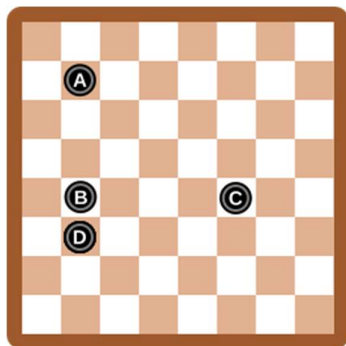




### SOROK ÉS OSZLOPOK (2018-BE-03)

SENIOR – NEHÉZ

JUNIOR – NEHÉZ



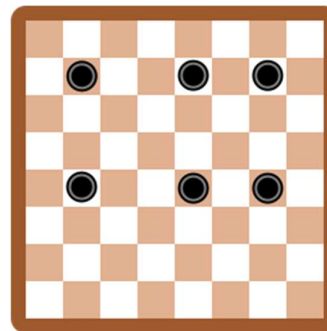
A jobb oldali diagramot a játéktábla alapján a következőképpen készítettük:

- Minden követ egy körrel ábrázolunk.
- 2 követ összekötünk egy vonallal, ha a játéktáblán ugyanabban a sorban vagy ugyanabban az oszlopban találhatók.

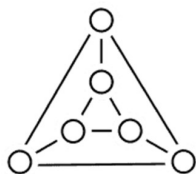
A köveken és a diagram körében szereplő betűk segítenek az ellenőrzésben.

Az alábbi játéktáblán 6 kő található:

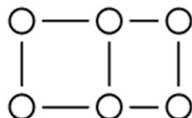
Melyik diagram ábrázolja a 6 köves játéktáblánkat?



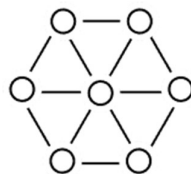
A.



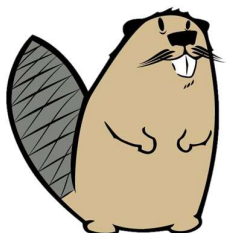
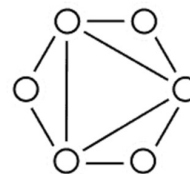
B.



C.



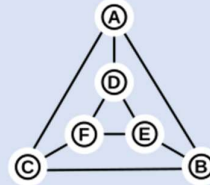
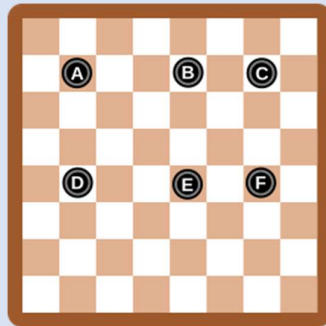
D.





### „A” válasz a helyes

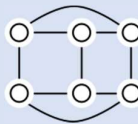
A következő képen, melyen a köveket betűkkel jelöltük, jól visszakövethető a megoldás:



A B, C és D diagramokat a következő okokból zárhatjuk ki:

Minden kővel két másik található egy sorban és egy kő ugyanabban az oszlopban. Ez azt jelenti, hogy minden követ pontosan  $2+1=3$  másik kővel kell összekötni. A többi diagram esetében ez nem így van, ráadásul a C diagram esetében 7 kör szerepel.

A B diagram becsapós, hiszen hasonlít a játéktáblára. De a szélső 4 kör csak 2 másik körrel van összekötve. Ahhoz, hogy megfelelő legyen, további két vonalat még be kell rajzolnunk:



### MIÉRT INFORMATIKA?

Az informatikában ilyen diagramokat sokszor használnak azért, hogy egy-egy probléma lényeges részét ábrázolják. Az ilyen diagramokat gráfnak hívják. A köröket csomópontnak, a vonalakat éleknek nevezik.

A gráfok esetében csak az lényeges, hogy melyik csomópont melyik más csomópontokkal van összekötve az élekkel. A csomópontok elrendezése, az élek formája nem játszik semmilyen szerepet. Tehát ugyanaz a gráf különbözőképpen is lerajzolható, mint láthattuk: az A megoldás és a javított C megoldás is megfelelő és ugyanazt a gráfot ábrázolja.

A gráf az absztrakció (elvonatkoztatás) formája – egy probléma lényegi részét ábrázolja. Ennek a hód-feladatnak az esetében a gráfunk segítségével például megválaszolhatunk olyan kérdéseket, mint „mennyi az a legkevesebb kő, amelyet el kell távolítanunk a tábláról ahhoz, hogy ne legyen egy sorban, vagy egy oszlopban két kő.

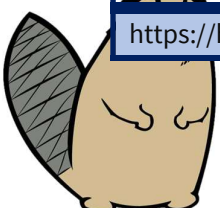
Egy informatikus munkájának lényeges részét képezi egy probléma esetén a megfelelő ábrázolás (reprezentáció) megtalálása, mely a megoldásban is sokat segíthet.

### KULCSSZAVAK

Gráf, Adatábrázolás

### WEBOLDALAK

<https://hu.wikipedia.org/wiki/Gráf>





**PRÓBATERV (2018-BE-04)****SENIOR – NEHÉZ**

Egy fellépésre készül öt táncos: Alex, Bojan, Coco, Deniz és Emil.

A fellépésen párokat alkotnak:

- Alex - Bojan
- Coco - Alex
- Emil - Deniz
- Alex - Emil
- Coco - Deniz
- Bojan - Coco
- Deniz - Alex
- Coco - Emil

Holnap a párok egymás után próbálnak. Ehhez a tánctanár a következő kitételek alapján akarja a sorrendet felállítani:

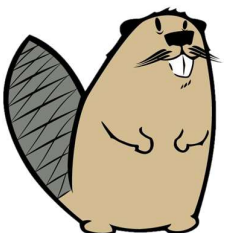
1. senki sem táncolhat háromszor egymás után<sup>1</sup>, és
2. a táncoló páros egyik tagja legyen benne a következőként táncoló párosban. Például az Alex-Bojan páros tánca után egy olyan pár következzen, melyben vagy Alex, vagy Bojan benne van – azaz Coco-Alex, Alex-Emil, Bojan-Coco vagy Deniz-Alex.

A táncosok egyike holnap később is érkezhetsz a próbára: ha a tánctanár kitételei teljesülnek, akkor sem lehet semmiképpen az első táncoló párosban.

**Melyik táncos lehet ez?**

---

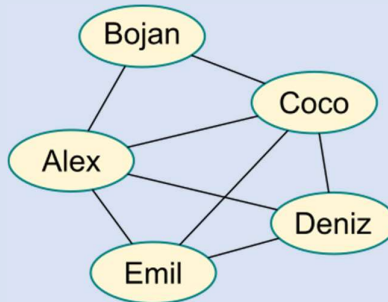
<sup>1</sup> A 2018-as versenyen ez a kitétel még nem szerepelt. De a feladat egyértelműsége megköveteli!





## „Bojan” a helyes válasz

A képen minden táncos nevét beírtuk egy oválisba. Két oválist összeköttöttünk egy vonallal, ha egy párt alkotnak. Egy próbaterv, melyben a tánctanár kitételei teljesülnek, tulajdonképpen egy „út” a képen: az egyik oválisnál kezdődik és minden vonalon pontosan egyszer megy át.



Ehhez minden oválist, amelyhez eljutottunk, el is kell hagynunk egy másik „úton”. Ebből következik, hogy ebből az oválisból páros számú vonalnak kell „kiágaznia”. Ez alól kivétel az az ovális, mellyel az út kezdődik, illetve amelyiknél végződik – ezekhez páratlan számú vonal csatlakozik.

Csak Deniz és Emil nevét tartalmazó oválishoz csatlakozik páratlan számú vonal (szám szerint 3). Tehát csak ők ketten lehetnek az első (vagy az utolsó) párosban. Bojan az egyetlen táncos, aki sem Denizzel, sem Emillel nincs párban, ezért ő az egyetlen, aki biztosan nem kezd a tervek szerint.

## MIÉRT INFORMATIKA?

A megoldásban látható kép megmutatja nekünk, hogy a táncos párokat hogyan ábrázolhatjuk gráf segítségével. Egy gráf csomópontokból (táncosok) és élekből (táncosok párosítása) áll. Ez egy igen sokoldalú struktúra, melyet sok modellezéshez köthető informatikai problémafelvetésben pl. (közlekedési, kommunikációs, ...) hálózatoknál használnak. Ebben a hód-feladatban a táncosok pár-hálózatot alkotnak.

Sok hálózatban fontos, hogy meghatározzuk egy kiinduló csomópontból egy (másik, különböző) célcsomópontba való összes eljutás lehetőségét. Ehhez felmerül (pl. hatékonysági okokból) a kérdés, hogy létezik-e olyan út, ahol minden élen (összekötésen) csak egyszer megyünk végig. Amennyiben ez lehetséges, a megtalált utat Euler-útnak is nevezzük. A gráfelmélet atyja, Leonhard Euler bizonyította, hogy ez csak akkor lehetséges, ha pontosan két páratlan kapcsolattal (élel) rendelkező csomópontunk van (és az összes többi csomópont páros számú élel rendelkezik). Csak ezek a csomópontok lehetnek egy Euler-út kiinduló és végpontjai.

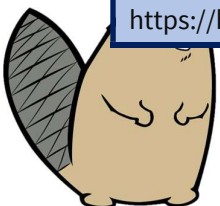
Ha ismerős számodra ez a hód-feladat, az nem véletlen. Keresd meg az „elődjét”, melyben a gráf kicsit más irányban áll.

## KULCSSZAVAK

Gráf, Adatábrázolás

## WEBOLDALAK

<https://hu.wikipedia.org/wiki/Gráf>





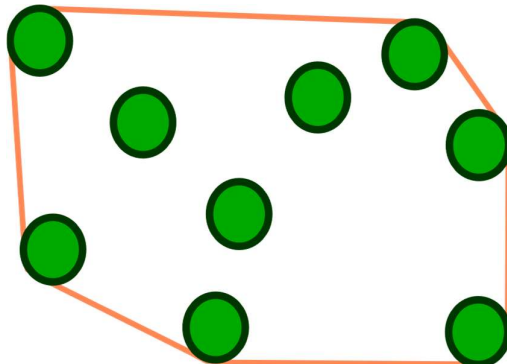
**FÁK EGY KÖRBEN (2018-CA-02)**

KISHÓD – KÖZEPES

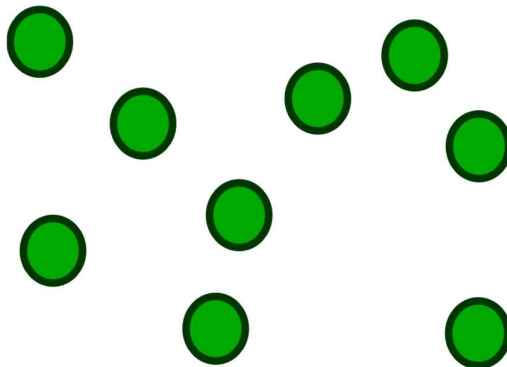
BENJAMIN – KÖNNYŰ

A hódok mindig egy hosszú szalaggal kerítik körbe azokat fákat, amiket ki akarnak vágni.

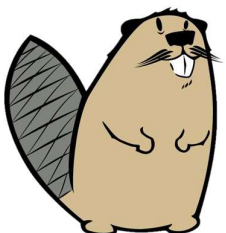
Tegnap hat fát akartak kivágni. De a szalag csak öt fát érintett, ami a levegőből így nézett ki:



Ma a hódok a következő fákat szeretnék kivágni:



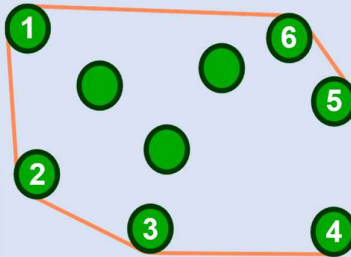
Hány fát érint most a kihúzott szalag?





## „6 fát” a helyes válasz

A hódok a következőképpen keríthetik körbe a fákat:



A szalag a számokkal jelölt hat fát érinti.

## MIÉRT INFORMATIKA?

A hód-fák-szalag a legkisebb területet keríti el, amelyben az összes kivágandó fa van. Ha a fák a megoldásban szereplő képen apró pontok volnának, akkor a szalag egy hatszögű forma lenne.

Egy ilyen legkisebb sokszöget, ami egy adott halmazból minden pontot tartalmaz, matematikában „konvex buroknak” neveznek. Ahol a „konvex” azt jelenti, hogy valami kifelé domború, mint például egy nagyító konvex lencséje. A „burok” pedig valami, ami valami mást magába zár, mint a bőr a testet, de nem nagyobb területen, mint ami épp szükséges. A hód-fák-szalag is olyan, mint a konvex burok: körbekeríti az összes fát és nem megy a fák közé – és a „nem megy közé” a matematikában annyit jelent, mint a „kívül”.

Az informatikában gyakran fontos, hogy a pontok halmazának konvex burkát ki tudjuk számítani:

- Mintafelismerés: A képen egy arc látható?
- Kézírásfelismerés: Ez a kézzel írt karakter a B betű?
- Földrajzi információs rendszerek: Milyen nagy egy ártér vagy egy folyórendszer?
- Csomagolás: Mi a legkisebb mennyiség, ami elég egy bizonyos tárgy becsomagolásához?

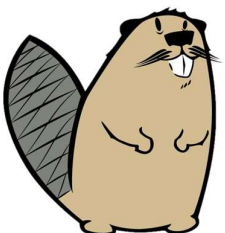
Az informatika ismer olyan eljárásokat, melyekkel egy ponthalmaz konvex burkát eredményesen tudja kiszámolni. Ezek akkor is jól működnek, ha a ponthalmaz nagyon nagy.

## KULCSSZAVAK

Konvex burok, Algoritmus

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Konvex\\_burok](https://hu.wikipedia.org/wiki/Konvex_burok)

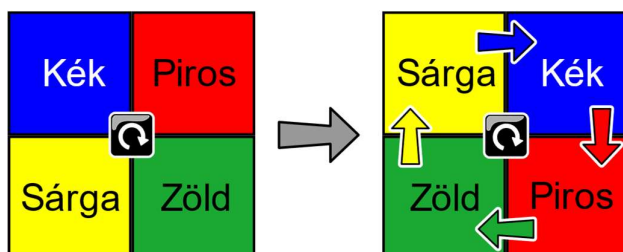




## SIMON MONDJA (2018-CA-03A)

KISHÓD – KÖNNYŰ

Amikor Simon megnyomja a középső gombot, a következőképpen mozdulnak el a négyzetek:



Simon még kétszer megnyomta a középső gombot. Hogy helyezkednek el a négyzetek?

Hogy helyezkednek el a négyzetek?

A.



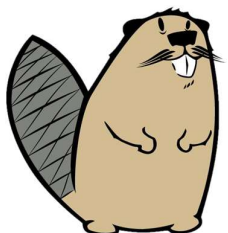
B.



C.

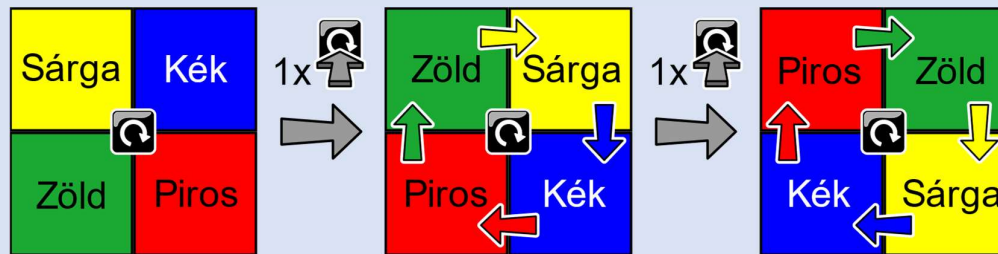


D.





## „A” válasz a helyes



## MIÉRT INFORMATIKA?

A feladat egy gépet ír le, aminek a négy pozíciója mindig egy bizonyos állapotban van: piros, zöld, kék és sárga. Egy gombnyomás minden pozíció állapotát megváltoztatja ebben a sorrendben:

piros → kék → sárga → zöld → piros.

Egy ilyen gépet nevezünk az informatikában véges automatának. Az emberek szívesebben tűnődnek azon, hová kerülnek a színek, tehát hogy az órajárással megegyező irányba mozognak-e.

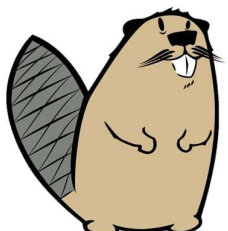
Sok játék van, ahol bizonyos sorrendet kell követni. A „Simon mondja” nevű játék is éppen ilyen. A játékvezető utasításokat ad a játékosoknak, amiket követniük kell, ha a játékvezető „Simon mondja”-t mond. Ha például azt mondja, hogy „Mindenki nyújtsa ki a nyelvét!”, akkor senkinek nem kell semmit sem csinálnia. Viszont, ha azt mondja „Simon mondja, mindenki ugráljon egy lábon!” akkor mindenkinek muszáj egy lábon ugrálnia. Aki hibázik, kiesik.

## KULCSSZAVAK

Véges automata, gép, utasítások, algoritmus

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Determinisztikus\\_véges\\_állapotú\\_gép](https://hu.wikipedia.org/wiki/Determinisztikus_véges_állapotú_gép)





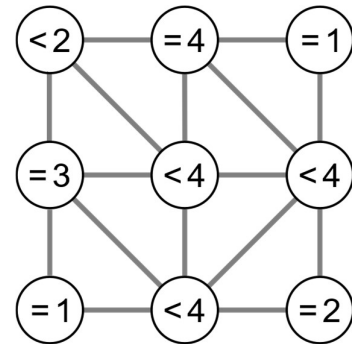
## SZOMSZÉDOK (2013-AT-11)

KADÉT – NEHÉZ

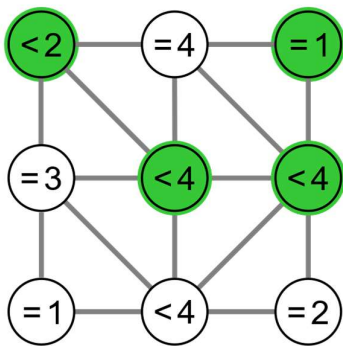
JUNIOR – KÖZEPES

Az alábbi képen kilenc kör látható, amelyek részben össze vannak kötve.

Egy összekötés szomszédá teszi őket. A körök közül kell kiválasztanunk néhányat. Minden körben áll egy kifejezés, ami azt mutatja meg, hogy hány kiválasztott szomszédjának kell lennie.



„=3” azt jelenti, hogy pontosan három kiválasztott szomszédja van. „<4” pedig, hogy maximum három kiválasztott szomszédja van.

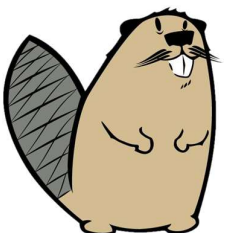


Már előre megjelöltünk néhány kiválasztott kört.

Ahhoz, hogy mind a kilenc körben egyszerre teljesüljenek a kifejezések, **még egy kört** kell kiválasztani.

Melyiket?

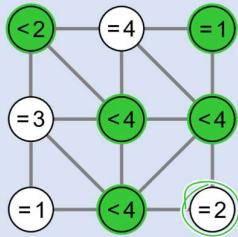
- A. A „<4”-t tartalmazó kört.
- B. Az „=3”-t tartalmazó kört.
- C. Az „=2”-t tartalmazó kört.
- D. Az „=1”-t tartalmazó kört.



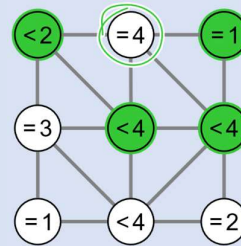


## „A” válasz a helyes

A felül közepén lévő körnek („=4”) pontosan négy szomszédja van, szóval mind a négyet ki kell választanunk.



Éppen így a jobboldali „=2”-t tartalmazó körnek is pontosan két szomszédja van, szóval mindkettőt ki kell választanunk.



(A középső sor jobb oldali köre „=2” felső szomszédja már ki van választva)

Ezzel már az összes kör minden feltétele teljesül:

Ha még egy kört kiválasztanánk, a feltételt megsértenénk:

- Ha „=4” kört fent közepén kiválasztanánk, akkor a „=1” kör feltétele fent jobb oldalon nem teljesülne.
- Ha „=3” kört bal oldalt közepén kiválasztanánk, akkor a „<2” kör feltétele fent bal oldalon nem teljesülne.
- Ha „=1” kört bal oldalt alul kiválasztanánk, akkor a „=3” kör feltétele középen bal oldalon nem teljesülne.
- Ha „=2” kört lent jobb oldalt kiválasztanánk, akkor a „<4” kör feltétele középen jobb oldalon nem teljesülne.

## MIÉRT INFORMATIKA?

Mennyi próbálkozásra van szükség egy probléma megoldásához? Ha egyszerűen minden lehetséges megoldást kipróbálunk, akkor  $2^9 = 512$  különböző lehetőség van. Amikor minden lehetőséget kipróbálunk, azt brute-force-módszernek nevezzük. Majd ellenőrizni kell ezeket a lehetséges megoldásokat, hogy a feltételek teljesülnek-e bennük.

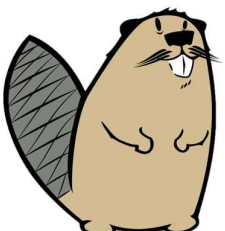
Érdekes azonban ebben az esetben logikusan és konzekvensen eljárni. Először azokat a köröket keressük meg, amikre a feltétel egyértelműen teljesülhet. Például minden kör, amelynél a feltétel „=n” és pontosan n kapcsolata, szóval n szomszédja van. Innen tudunk majd logikus következtetéssel továbbmenni: megnézni a nem teljesült feltételeket, amik csak ily módon teljesülhetnek. Így tudunk majd sok kis lépéssel a megoldáshoz elérni. Általános esetben meg lehet azt is állapítani, hogy nincs megoldás (vagy legalábbis megtalálhatunk egyet a sok lehetőség közül). Egy ilyen analitikus eljárásmodot, aminél minden megoldási lehetőség közül csak az ígéreteseket nézzük meg, heurisztikusnak hívunk.

## KULCSSZAVAK

Logika, gráf, szomszédság

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Szomszédság\\_\(gráfelmélet\)](https://hu.wikipedia.org/wiki/Szomszédság_(gráfelmélet))





## TÁRSASJÁTÉK (2018-CA-06)

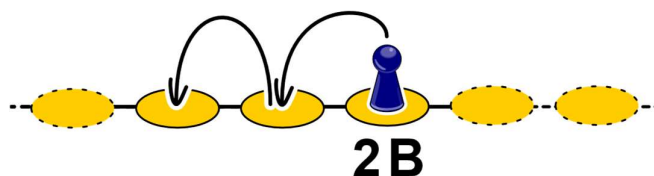
JUNIOR – KÖZEPES

SENIOR – KÖNNYŰ

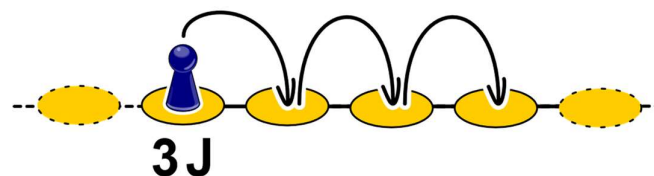
Mint minden társasjátékban, itt is az egyes mezőkre csak bizonyos szabály szerint lehet ráugrani.

Ennél a játéknál minden mezőhöz tartozik egy szabály. Háromféle szabály van:

**nB:** n mezőt ugrunk balra, 2B azt jelenti tehát, hogy 2 mezőt ugrunk balra:

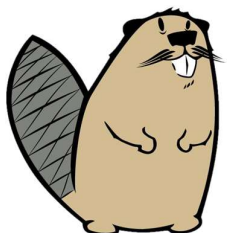
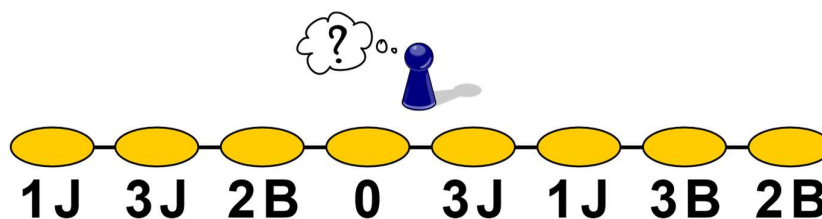


**nJ:** n mezőt ugrunk jobbra, 3J így azt jelenti, hogy 3 mezőt ugrunk jobbra:



**0:** nem lehet tovább ugrani

Melyik mezőn kell kezdeni ahhoz, hogy a játék végén minden mezőn egyszer járjunk?

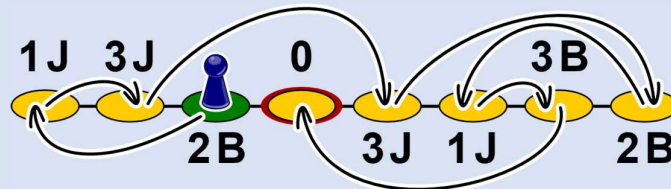




## „Balról a harmadik mezőn (2B)” a helyes válasz

Ha balról a harmadik mezőn („2B”) kezdünk, akkor a játék végére minden mezőre csak egyszer lépünk.

A „0” mezőtől visszafelé kell gondolkodni: melyik mezőről érjük el a „0” mezőt. Ebben az esetben ez jobb oldalról a második mező, azaz a „3B”. Ezt a mezőt pedig a jobbról a harmadik mezőről lehet elérni (1J), amit egészen a jobb oldalon lévő mezőről (2B), amit jobbról a negyedik mezőről (3J), majd balról a második mezőről (3B), amit egészen a bal oldalt lévő mezőről (1J) lehet elérni és végül balról a harmadik mező maradt (2B).



Az ábrán lévő nyilak mutatják a lépések útját, ezek egy irányított gráfot alkotnak. A „0” mezőről indulva csak visszafelé kell lépdelni és akkor megkapjuk a kiindulási állapotot, mely jelen esetben balról a harmadik mező.

## MIÉRT INFORMATIKA?

Az informatikában hasonlóan működik a „láncolt listák” tárolási elve, mint az ugrálás egyik mezőről a másikra: a munkamemóriában az objektumok a következő memóriacímére utalnak. Így egy objektum a memória tetszőleges helyén eltárolható és nem kell időigényes keresést vagy átpakolgatást végezni ahhoz, hogy egy egybefüggő helyre teheszük. A programozónak ezzel nem kell törődnie, csak azzal, hogy megfelelő nagyságú memóriával rendelkezzen.

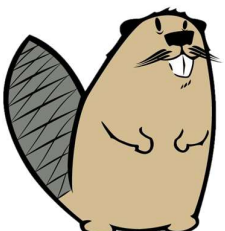
De mi történik akkor, ha az objektumra már nincs szükség a memóriában? Míg régebben a programozóknak kellett azzal törődniük, hogy a memóriát újra felszabadítsák (ami gyakran sajnos más irányba ment, így a program idővel egyre több memóriát használt fel), ma már a modern programnyelvek képesek arra, hogy a feleslegessé vált dolgoktól megszabaduljanak. A „Garbage Collection” rendszeresen ellenőrzi, hogy az objektumra a memóriában még hivatkozik-e másik objektum (hogy még utal-e rá valami). Néha az objektumok nagy struktúrája már nem mutat semmire sem, akkor ilyenkor mint a példában: vissza kell követni, hogy melyik az a kiindulási pont (startmező), amire még a másik objektum utal. Ha ....

## KULCSSZAVAK

Memória, algoritmus, szemétgyűjtés

## WEBOLDALAK

<https://hu.wikipedia.org/wiki/Szemétgyűjtés>





**SZÁMÍTÓGÉPES JÁTÉK (2018-CH-03)****BENJAMIN – NEHÉZ****KADÉT – KÖZEPES**

Andrea egy számítógépes játékot programozott az iskolában. A játékszabály igazán egyszerű:

A játék egy sor játékkörből áll. Egy játékkörben leesik egy levél. A hód megpróbálja a levelet elkapni, mielőtt az földet érne. Ahhoz, hogy a hód nyerjen, 15 levelet kell elkapnia, mielőtt 4 levél földet érne.

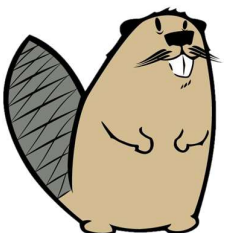
A játék hossza függ a játékkörök számától.

A következő példában a hód 6 játékkör után veszít, mert a maximum 4 nem elkapott levelet elérte. A játék hossz ebben a példában 6 levelet tesz ki.

Játékkör	Eredmény	Pontszám – Lapok száma	
		Elkapott	Nem elkapott
1. Játékkör	elkapta	1	0
2. Játékkör	nem kapta el	1	1
3. Játékkör	elkapta	2	1
4. Játékkör	nem kapta el	2	2
5. Játékkör	nem kapta el	2	3
6. Játékkör	nem kapta el	2	4

**Meddig (hány játékkörig) tart a leghosszabb játék?**

- A. 4 játékkör
- B. 15 játékkör
- C. 18 játékkör
- D. nincs korlát





### „C” válasz a helyes

A leghosszabb lehetséges játék megtalálásához, az összes olyan helyzetet kell kombinálnunk, amiben a játék továbbmegy. Ehhez a maximálisan elkapott leveleket (15 játékkör) kombináljuk a maximálisan nem elkapott levelekkel (3 játékkör). A nem elkapott maximális levelek száma a 3, mert a 4 már megállítaná a játékot. Ha a játék 4 nem elkapott levélhez vezetne, akkor a maximum elkapott levelek száma 14 lenne, különben a játék megint véget érne. Ezért a maximális hossz  $15 + 3 = 14 + 4 = 18$  kör és a helyes megoldás a C.

Az A megoldás – 4 kör – a legrövidebb hossza a játéknak (ha egyik levelet sem kapja el).

A B megoldás a játék legrövidebb hossza, amivel a játékot megnyeri (minden levelet megfog)

A D megoldás rossz, mivel az elkapott levelek maximumát vagy a nem elkapott levelek maximumát már korábban elérte volna.

### MIÉRT INFORMATIKA?

Egy játék programozásánál a szabályokat egyértelműen kell meghatározni (definiálni).

A szabályok hatásainak is egyértelműnek kell lennie. A játék olyan módon legyen felépíthető, hogy a nyereség és veszteség is lehetséges legyen (elegendő levél álljon a rendelkezésre) és a játék ne legyen se túl rövid, se túl hosszú.

Egy játék, ami több körből áll, egy folyamat. Az informatikusok a folyamatok (processzek) modellezésének és leírásának szakértői. Egyik fő feladatuk az, hogy kitalálják, mi minden történhet és mennyi ideig futhat a folyamat. Ha ....

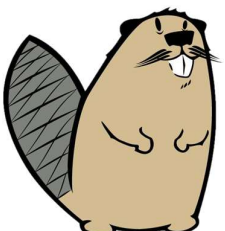
### KULCSSZAVAK

Folyamatok, programozás

### WEBOLDALAK

[https://hu.wikipedia.org/wiki/Bug\\_\(informatika\)](https://hu.wikipedia.org/wiki/Bug_(informatika))

[https://hu.wikipedia.org/wiki/Program\\_\(informatika\)](https://hu.wikipedia.org/wiki/Program_(informatika))





BARÁTLÁTOGATÁS (2018-CH-05)

KADÉT – NEHÉZ

JUNIOR – KÖZEPES

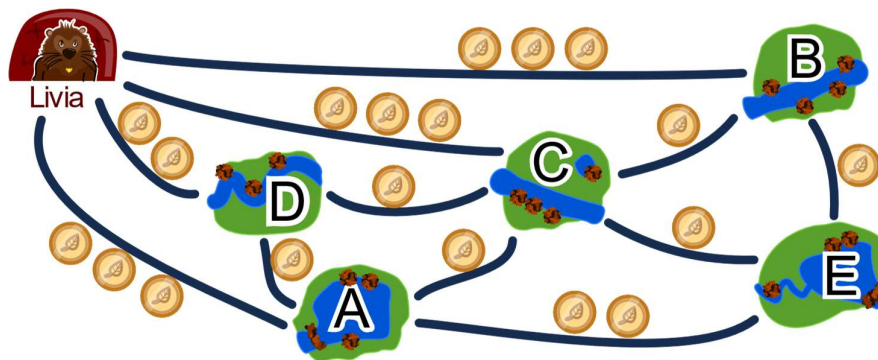
SENIOR – KÖNNYŰ

Lívia meg szeretné látogatni a barátait az A, B, C, D és E betűkkel jelzett falvakban.

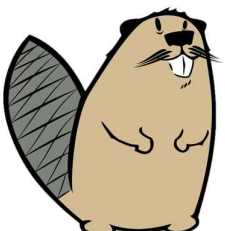
Mindenkit egy utazással látogat meg úgy, hogy nem megy el egynél többször ugyanabba a faluba és az utazás végén hazaér. Az egyes útszakaszok árát a képen láthatod.

Az egyik lehetséges útvonal, hogy meglátogassa a barátait 11 hódpénzbe kerül:

Lívia háza → B → E → A → D → C → Lívia háza



Vajon mennyibe kerülne az utazása, ha a legolcsóbban (a legkevesebb hódpénzből) szeretne kijönni?





**„9 hódpénz” a helyes válasz**

Két optimális, azaz legolcsóbb lehetősége van:

Lívia háza  $\rightarrow B \rightarrow E \rightarrow C \rightarrow A \rightarrow D \rightarrow$  Lívia háza

Lívia háza  $\rightarrow D \rightarrow A \rightarrow C \rightarrow E \rightarrow B \rightarrow$  Lívia háza

A két megoldás egymás fordítottja és mindkét esetben 9 hódpénzt kell Líviának fizetnie.

Ennél jobb (olcsóbb) megoldást nem tudunk találni, mivel Lívia házától egy 2 hód pénzbe kerülő út vezet ki. A másik három út már 3-3 hódpénzbe kerül. Mivel hazafele másik úton kell mennie („nem megy el egynél többször ugyanabba a faluba”), ezért 5 hód pénznél tartunk. A hátralévő falvak meglátogatását utanként 1-1 hódpénzzel megoldhatjuk. Ami a 4 út miatt így már összesen 9 hód pénz.

Az összes többi megoldás ennél többbe kerül.

**MIÉRT INFORMATIKA?**

Egy jó, akár optimális megoldás megkeresése az informatikában alapvető feladat.

Ezt a hód feladatunkat ábrázolhatnánk úgynevezett gráffal is, ahol a barátok lakhelyei a csomópontok, az utak pedig az élek. A feladatunk pedig minden csomópont „bejárása” pontosan egyszer úgy, hogy az élek (utak) a legkevesebb összsúllyal (költséggel) rendelkezzenek. Ehhez hasonló az úgynevezett „Utazó ügynök probléma” (angolul: Traveling Salesman Problem (TSP)).

Az ilyen típusú problémákat általában nagyon nehéz megoldani számítógéppel. A megoldások alapja az, hogy az összes lehetséges megoldást kipróbáljuk és közben figyeljük, melyiknél mennyi a költség. Ezt lerövidíthetjük, ha valamilyen stratégiából indulunk ki. Pl. hogy mindig a legrövidebb utat választjuk ki. Majd ehhez hasonlítjuk a további bejárásainkat.

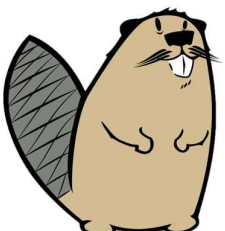
Ha egy-egy csomópontba nem csak egyszer mehetünk, a feladat tovább bonyolódik, hiszen még több lehetőségünk lesz a megoldásra.

**KULCSSZAVAK**

Optimalizáció, Utazóügynök probléma

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Az\\_utazó\\_ügynök\\_problémája](https://hu.wikipedia.org/wiki/Az_utazó_ügynök_problémája)



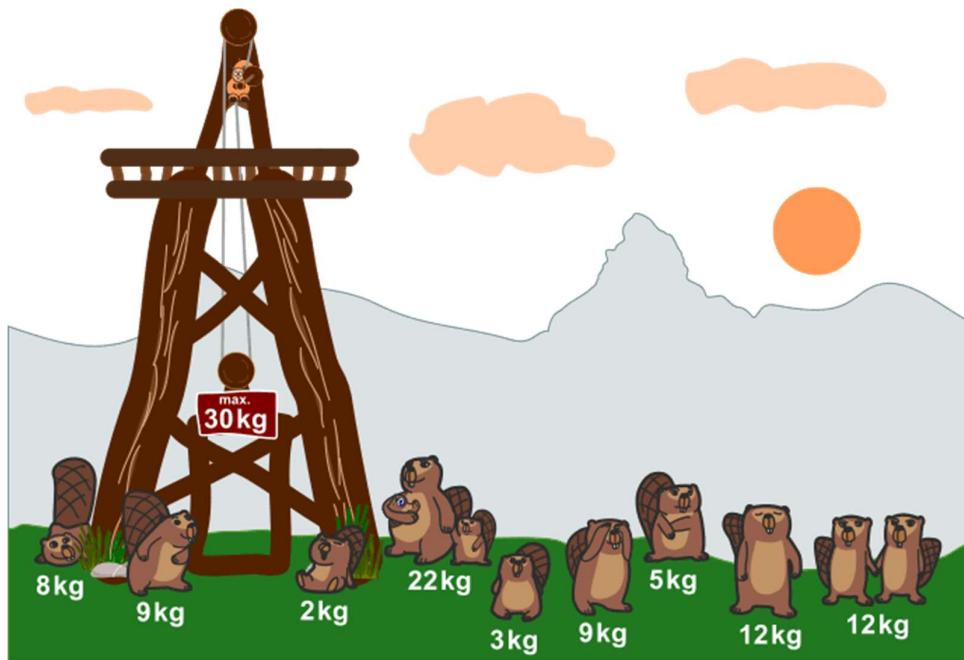


### FELVONÓ A MATTERHORN-ON (2018-CH-07)

BENJAMIN – NEHÉZ

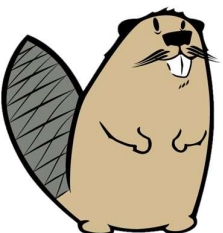
KADÉT – KÖZEPES

A hódcsalád egy kilátóhoz kirándult. Elég későn értek oda, így a lift már csak kétszer mehetett fel. A lift legfeljebb **30 kg**-ig terhelhető. A kisebb hódok nem mertek felmenni az anyukájuk nélkül a toronyba. Az ikrek is csak együtt akartak felmenni.



Hány kilogramm terheléssel menjen fel a lift az egyes fordulókban, hogy a lehető legtöbb hód élvezhesse a kilátást?

- A. Mindkét fordulóban 30-30kg.
- B. Az egyik fordulóban 27kg, a másikban 24kg.
- C. Az egyik fordulóban 30kg, a másikban 27kg.
- D. Az egyik fordulóban 27kg, a másikban 21kg.

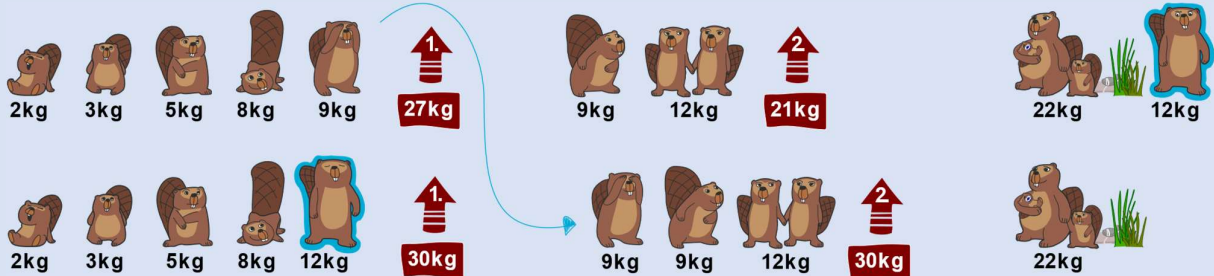




## „A” válasz a helyes



Egy lehetséges megoldás lehet, ha a lehető legtöbb „legkönnyebb” hód száll be a liftbe:  $2+3+5+8+9=27\text{kg}$  először, majd  $9+12=21\text{kg}$  (ez 8 hódot jelent). De még egy hód befér:



Ezzel a stratégiával mindkét szállításnál optimálisan kihasználjuk a helyet: az első csoportba sorolt 9kg-os hód helyett a „kimaradt” 12 kg-os hód szállna be. Így az első menetben kihasználjuk az maximum 30kg-os lehetőséget. A 9 kg-os hód a második körben csatlakozik, akkor az a lift is „teltházás” lehet.

Nem ez az egyetlen optimális megoldás.

A kérdés az, hogy az így kimaradt 3 hód helyett meg tudjuk-e oldani azt, hogy csak 2 vagy 1 maradjon lent. Az első lift esetében három hódot ( $12+8+2=22\text{kg}$ ) kellene kiszállítani, hogy a mama hód a kis hódokkal beszállhasson. Ez ugyanúgy 3 hódot jelentene. A második körben mindenkinek ki kellene szállni ahhoz, hogy a mama hód a kis hódokkal beszállhasson, mivel  $9+9=18$  (kevés) és  $12+9=21$ , ami szintén kevés lenne.

## MIÉRT INFORMATIKA?

Egy probléma optimális megoldásának megtalálása klasszikus informatikai probléma.

Gyakran az ilyen megoldások nem találhatók meg elég gyorsan vagy meghatározható időn belül, mivel túl sok lehetséges megoldás létezik, melyet meg kell vizsgálni. Ezeket gyakorlatilag megoldhatatlan problémának is nevezik az informatikában.

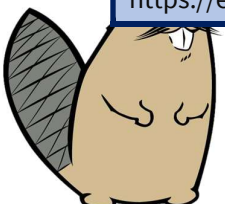
Ezt a problémát „Hátizsák-problémának” is nevezik, mivel meglehetősen sok objektumot kell bepakolnunk anélkül, hogy a maximális súlykorlátot elérnénk. Az ilyen és ehhez hasonló problémákat NP-teljesnek hívják. Csak megközelítőleg megoldhatóak, ami azt jelenti, hogy egy lehetséges és jó megoldást találhatunk, de nem szükségszerűen az optimálisat (legjobbat).

## KULCSSZAVAK

Optimalizálás, hátizsák probléma

## WEBOLDALAK

[https://en.wikipedia.org/wiki/Knapsack\\_problem](https://en.wikipedia.org/wiki/Knapsack_problem)




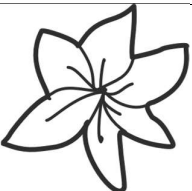

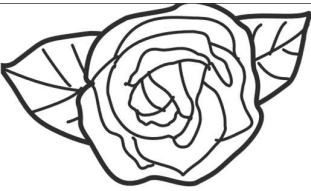


### KLÁRA VIRÁGAI (2018-CH-09)

KISHÓD – NEHÉZ

BENJAMIN – KÖZEPES

Klára szereti a virágcsokrokat. Elmegy egy virágboltba, ahol a következő fajta virágokat árulják:

Kardvirág	Liliom	Tulipán	Rózsza
			

Minden fajta virág a következő színekben kapható: fehér, kék, sárga

Klára szeretne egy hat virágból álló csokrot venni, amelyben

- Minden szín (fehér, kék, sárga) pontosan kétszer szerepel.
- Ugyanabból a fajtából való virágok nem szerepelnek ugyanolyan színben.
- Minden fajta virág legfeljebb kétszer szerepel.

Melyik virágcsokor felel meg Klára összes kívánságának?

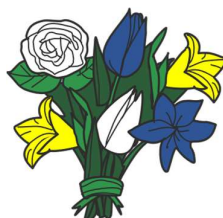
A.



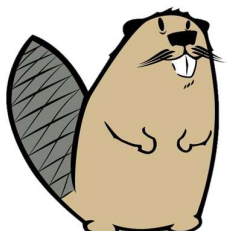
B.



C.



D.





**„D” válasz a helyes**

Az A csokorban három fehér virág szerepel (az 1. szabálynak mond ellent). A B csokorban három rózsza van (3. szabálynak mond ellent). A C csokorban két ugyanolyan fajtájú (kardvirág) virág ugyanazzal a színnel (sárga) szerepel (a 2. szabálynak mond ellent).

**MIÉRT INFORMATIKA?**

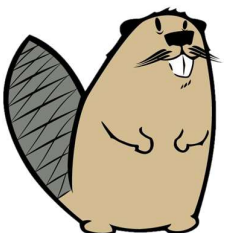
Az általános informatikai problémákat megszorítások sorával írjuk le és a feladat egy olyan megoldás megtalálása, amelyik minden, vagy legalább majdnem minden megszorításnak megfelel. Összetett feladatok esetében a megszorításokat logikai műveletek segítségével oldhatjuk meg. Ilyen logikai művelet az ÉS (amikor mindkét megszorításnak egyszerre kell teljesülnie), vagy a VAGY (amikor a két megszorításból legalább az egyiknek teljesülnie kell).

**KULCSSZAVAK**

Logikai műveletek, megszorítások

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Boole-algebra\\_\(informatika\)](https://hu.wikipedia.org/wiki/Boole-algebra_(informatika))





## RUHAKUPAC (2018-CH-10B)

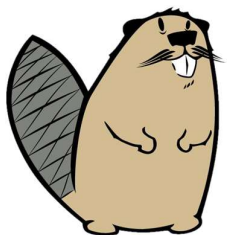
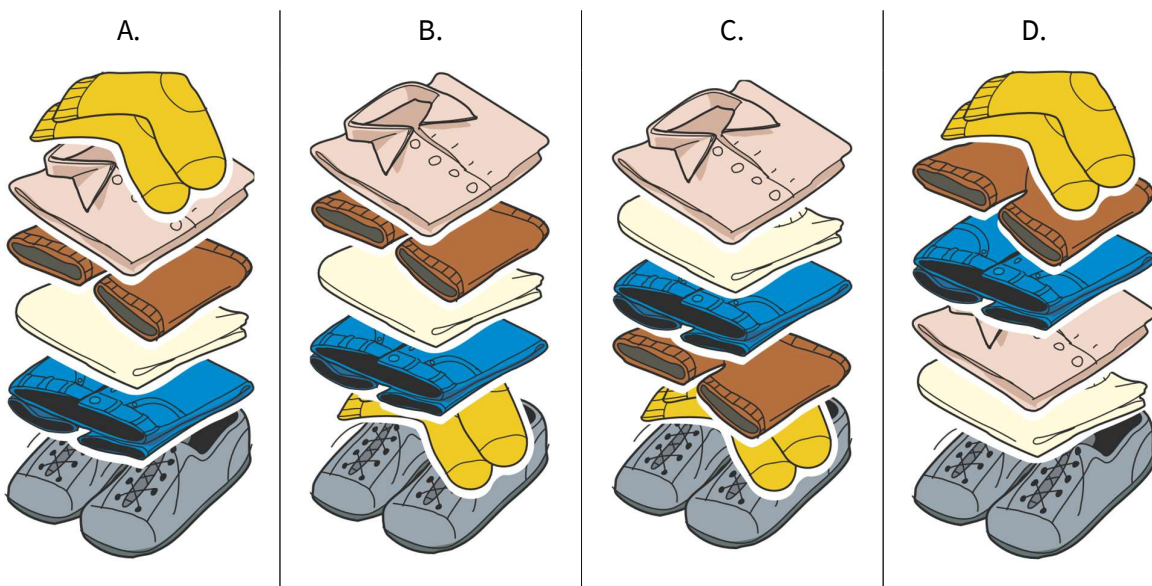
### KISHÓD – KÖNNYŰ

A hódmama gondosan előkészíti a ruhákat fiának, Brúnónak: egy kupacba hajtogatja azokat.



Brúnó olyan sorrendben veszi fel a ruhákat, ahogy édesanyja kikészítette neki: a ruhakupac tetejéről kezdve.

Vajon melyik ruhakupaccal NEM tud jól felöltözni?





**„C” válasz a helyes**

A C kupacban Brúnónak az alsónadrágot a nadrágra kellene felhúznia, de ő nem Superman 😊

A többi ruhakupac esetében az egymástól függő rétegek a helyes sorrendben kerülnek fel Brúnóra: a zokni a cipő előtt, az alsónadrág a nadrág előtt, az ing a pulóver előtt. Annak, hogy az alsónadrágot vagy az inget veszi-e fel először, nincs jelentősége.

**MIÉRT INFORMATIKA?**

„Mielőtt bemennél a szobádba, hogy az ott lévő játékdobozt felnyisd, ki kell nyitnod a szobád ajtaját.” Ez egy jó példa a rákövetkezésekre, kényszerítésekre: az ajtónak nyitva kell lennie ahhoz, hogy be tudj menni a szobába.

Annak vizsgálata, hogy egy előfeltétel adott-e ahhoz, hogy egy feladatot végrehajtsunk igen általános probléma az informatikában (is). Előfordul alkalmazások, számítógépes játékok esetében is.

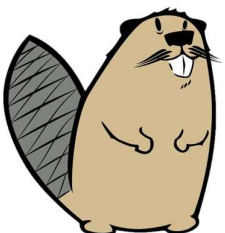
Ha csak bizonyos elemek esetében van megszorítás a rákövetkezésre akkor az olyan listákat (itt ruhakupacokat) részben rendezettnek nevezzük.

**KULCSSZAVAK**

Kényszerítések, parciális rendezés

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Részbenrendezett\\_halmaz](https://hu.wikipedia.org/wiki/Részbenrendezett_halmaz)





**KÖNYV CSERE-BERE (2018-CN-02)**

JUNIOR – NEHÉZ

SENIOR – KÖZEPES

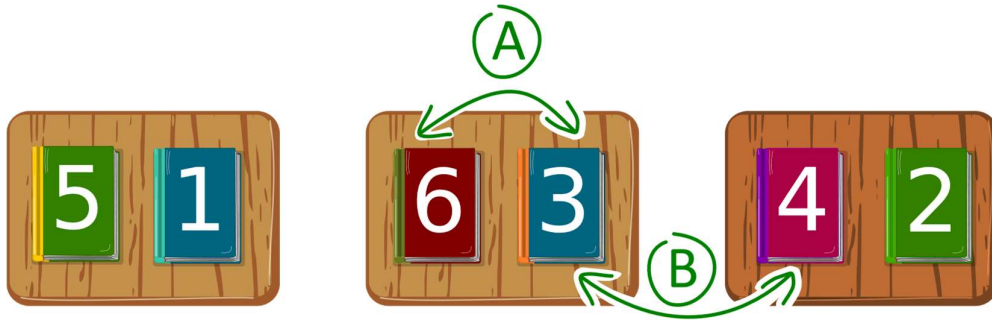
Három hód mindegyikének van egy asztala. Mindegyik asztalon két könyv.

Amint a képen is láthatod, a könyvek nincsenek sorrendben, de a hódok szeretnék sorba rendezni azokat. Ezt lépésenként teszik. Minden lépésben egy könyv csak egyszer mozdítható.

Két különböző lépés lehetséges, melyeket a hódok felváltva hajtanak végre:

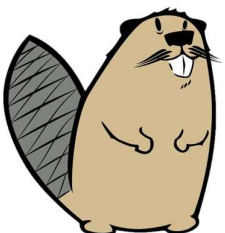
1. típus: minden hód megcserélheti (de nem kell feltétlen megcserélnie) az asztalán lévő két könyvet – ld. A példa
2. típus: a hódok minden könyvet egy szomszédos asztalon mellette lévő könyvvel megcserélhetnek (de nem kell feltétlen megcserélniük) – ld. B. példa

A képen látható, hogy helyezkednek el a könyvek kezdetben.



Első lépésként minden hód megcseréli a saját asztalán lévő két könyvet.

**Legfeljebb hány lépésre van ÖSSZESEN szükség ahhoz, hogy a könyvek sorrendben álljanak az asztalokon, azaz a sorrendjük 1, 2, 3, 4, 5, 6 legyen?**

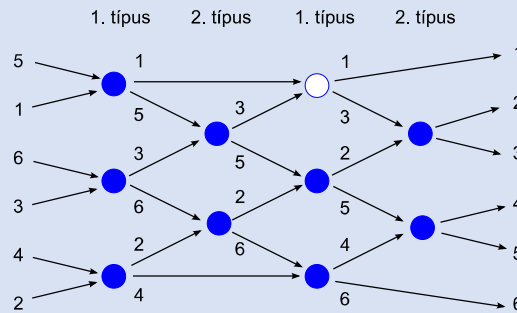




## „Négy lépésre” a helyes válasz

A kép azt mutatja, hogy az egyes cserék segítségével hogyan rendezhetjük a könyveket.

A hódok egy úgynevezett greedy-, vagy más néven mohó-algoritmust használnak. Ez azt jelenti, hogy minden lépésben egy kicsivel a végső megoldás fele akarnak eljutni. Összehasonlítják a szomszédos könyveket, amelyek abban a lépésben cserélhetőek lesznek. Ha ezek a könyvek már rendezettek (a baloldali könyvek kisebb a száma, mint a jobboldalinak), akkor nem csinálnak semmit. Különben kicserélik a két könyvet.



Az első lépésben (1. típus) minden asztalon megcserélik a könyveket.

A második lépésben (2. típus) a szomszédos asztalon, egymás mellett lévő könyveket cserélik meg.

A harmadik lépésben (1. típus) csak a két jobb oldali asztalon történik csere, és a negyedik lépésben (2. típus) minden szomszédos asztalon, egymás melletti könyv helyet cserél. Készen is vagyunk.

Gyorsabban (kevesebb lépésben) nem fog menni, mivel pl. az 5-ös könyvnek 4 hellyel kell jobbfelé kerülnie. Ehhez minimum 4 lépésre van szükség.

## MIÉRT INFORMATIKA?

Ebben a hód feladatban a sorbarendezés egy példa a párhuzamos algoritmusra. Több szereplő egy időben együtt old meg egy problémát.

A párhuzamos rendezés leírható egy rendezési hálóval, mint a megoldás esetében az ábra. Egy rendezési háló nyilakból és úgynevezett csomópontokból áll, melyeket most körökkel jeleztünk.

A nyilak a rendezni való elemeket ábrázolják. A csomópontok azt mutatják, hogy egy adott pillanatban mely objektumok kerülnek összehasonlításra és esetlegesen cserére.

Ha két csomópont egymás alatt helyezkedik el (egy oszlopban látszanak), akkor egyszerre (párhuzamosan) végrehajtható hasonlítás és cserét jelölnek. Ha követjük egy nyíl útját balról jobbra, felismerhetjük, hogyan kerül fokozatosan a megkívánt sorrendben lévő helyére az elem.

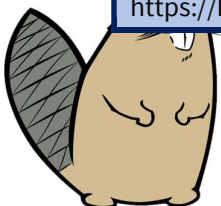
## KULCSSZAVAK

Párhuzamos rendezés, iteráció

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Párhuzamos\\_algoritmus](https://hu.wikipedia.org/wiki/Párhuzamos_algoritmus)

[https://hu.wikipedia.org/wiki/Mohó\\_algoritmus](https://hu.wikipedia.org/wiki/Mohó_algoritmus)





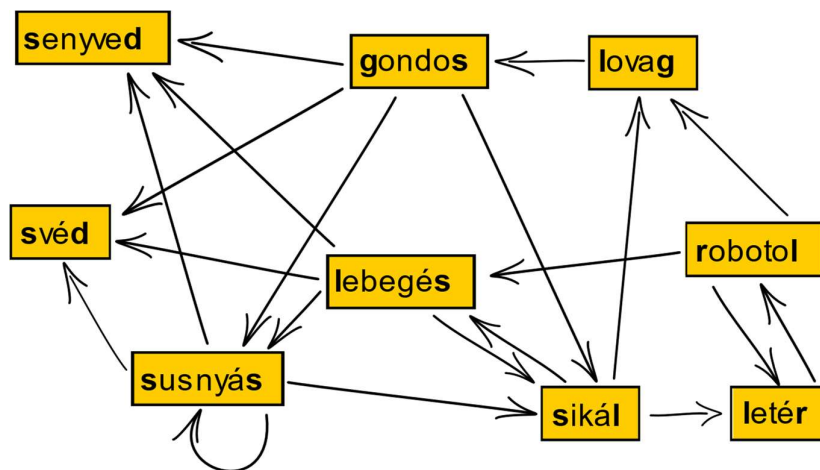
### A LEHETŐ LEGHOSSZABB SZÓLÁNC (2018-CZ-08C)

SENIOR – KÖZEPES

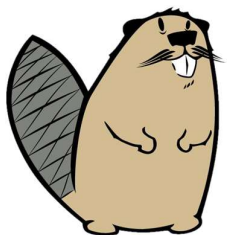
JUNIOR – NEHÉZ

A hódok egy szójátékot játszanak: az egyikük egy szóval kezdi. A következőnek egy olyan szóval kell folytatnia, amelyik az előző szó utolsó betűjével kezdődik. És ez így megy tovább.

Szavakat csak a megadott listáról választhatnak, és minden szó csak egyszer szerepelhet.



Milyen hosszú a leghosszabb szólánc, ami így létrehozható?





### „8 szó” a helyes válasz

Például a következő szólánc egy lehetséges megoldás:

lebegés – sikál – letér – robotol – lovag – gondos – susnyás – senyved

Találsz te is hasonló szóláncot?

De alkotunk-e hosszabb, azaz 8-nál több szóból álló láncot? A szóláncban 9 szó szerepel. Mivel mindegyiket csak egyszer használhatjuk fel, ez lenne a legtöbb. De nézzük meg a „senyved” és a „svéd” szavakat. Mivel nincs olyan szavunk, ami „d”-vel kezdődne, ezért a d-re végződő szónak kell a láncban az utolsónak lennie. Az egyiket a két szó közül, ami „d”-re végződik, nem tudjuk felhasználni. Így 8 lehet a maximális hossz.

### MIÉRT INFORMATIKA?

Ez a hód feladat az előre megadott szabályok és az ehhez kapcsolódó grafikus megjelenítés megértéséről szól. A leírásen belül egy optimális (lehető legjobb) megoldást kell megtalálnunk. Az ilyen feladatok az informatikusok mindennapjaihoz tartoznak.

Egy kis háttértudással felismerhetjük, hogy ez a feladat a leghosszabb út megkeresése egy irányított gráfban.

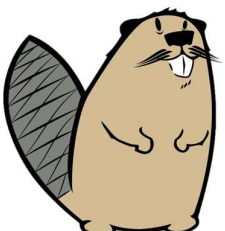
Ehhez a feladathoz hasonló játék az „Általánosított földrajz”, amikor városneveket rendezünk hasonlóan gráfba és a játékosok felváltva lépnek a nyilak mentén. Az veszít, aki már nem tud hova lépni.

### KULCSSZAVAK

Gráf, leghosszabb út, reprezentáció, dedukció

### WEBOLDALAK

<https://hu.wikipedia.org/wiki/Gráfelmélet#Út>  
[https://hu.wikipedia.org/wiki/Gráf#Irányított\\_gráf](https://hu.wikipedia.org/wiki/Gráf#Irányított_gráf)  
[https://en.wikipedia.org/wiki/Generalized\\_geography](https://en.wikipedia.org/wiki/Generalized_geography)













## Z Bolygó (2018-DE-02)

BENJAMIN – KÖZEPES

KADÉT – KÖNNYŰ

A Z bolygó lakói mindig ugyanúgy építik a városaikat. Minden várost egy házzal kezdenek. Azután az egyes objektumokat kicserélik az alábbi szabályok szerint:

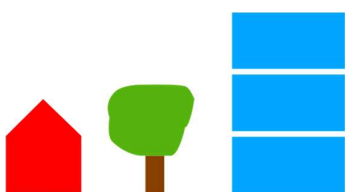


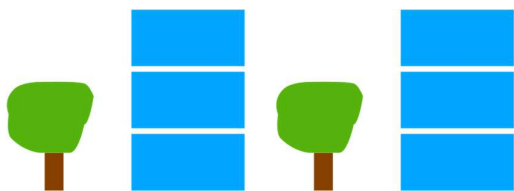
1.  →  
2.  →  
3.  → 

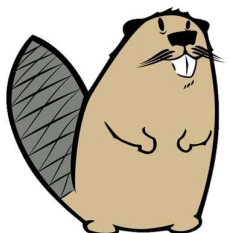
Például: ha először az 1., majd a 2. és végül kétszer a 3. szabályt alkalmazzák, egy olyan város alakul ki, mint a kép jobb oldalán látható:



Figyeld meg, hogy az egyes objektumok sorrendje nem változik.

Vajon melyik város NEM a Z bolygón található?

<p>A.</p> 	<p>B.</p> 
<p>C.</p> 	<p>D.</p> 





**„B” válasz a helyes**

Fák csak a 2. szabályt alkalmazva kerülhetnek a városba. De eszerint a szabály szerint a fa jobb oldalán (kék) téglalap(ok)nak kell állnia. A B városban a jobboldali fától közvetlenül jobbra nem áll téglalap. Ezért nem lehet a Z bolygón.

Az A város a következőképpen építhető fel: az 1., 2., 3. és végül ismét a 3. szabály alkalmazásával.

A C város az 1. szabályt háromszor egymás után alkalmazva építhető fel.

A D város építésekor először alkalmazzuk az 1. szabályt. Aztán minden házra alkalmazzuk a 2. szabályt és azután minden téglalapra a 3. szabályt kétszer.

**MIÉRT INFORMATIKA?**

Ez a hód feladat a helyettesítési szabályokról szól: egy objektumot más objektumok sorozatára cserélünk. Az informatikában az ilyen szabályokat környezetfüggetlennek nevezzük. Mivel egy objektum anélkül cserélődik valami másra, hogy fontos lenne a környezete (kontextusa, azaz ami tőle jobbra vagy balra helyezkedik el).

Az informatikában helyettesítési szabályokat alkalmazunk arra, hogy egy programozási nyelv szintaxisát meghatározzuk (definiáljuk). Ebben a feladatban a város házak, fák és téglalapok sorozata. Egy számítógépes program egy szöveg, ami betűkből, számokból és különböző egyéb jelekből áll. Ahogy a Z bolygón a városok a helyettesítési szabályokkal építhetők fel, a programok szövege is felépíthető: ahol egy kezdőszimbólummal kezdünk és helyettesítési szabályokat alkalmazunk. A kezdő szimbólum és a helyettesítési szabályok egy saját „nyelvtant” határoznak meg.

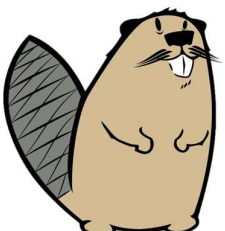
Minden programozási nyelvnek megvan a saját nyelvtana. Ez a nyelvtan határozza meg a programozási nyelv szintaxisát. Amikor a számítógép végrehajt egy programot, először azt vizsgálja meg, hogy a szöveg a programozási nyelven íródott-e (azaz a szintaxisát). Ehhez a programozási nyelv nyelvtanát használja.

**KULCSSZAVAK**

Helyettesítési szabály, formális nyelvek, nyelvtan

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Környezetfüggetlen\\_nyelvtan](https://hu.wikipedia.org/wiki/K%C3%B6rnyezetf%C3%BCggetlen_nyelvtan)





## AZ ELVESZETT AUTÓ (2018-DE-03)

BENJAMIN – NEHÉZ

KADÉT – KÖZEPES

Ilyen egy pechet! Egy önvezető autó nem ért haza, valahol a városban áll.

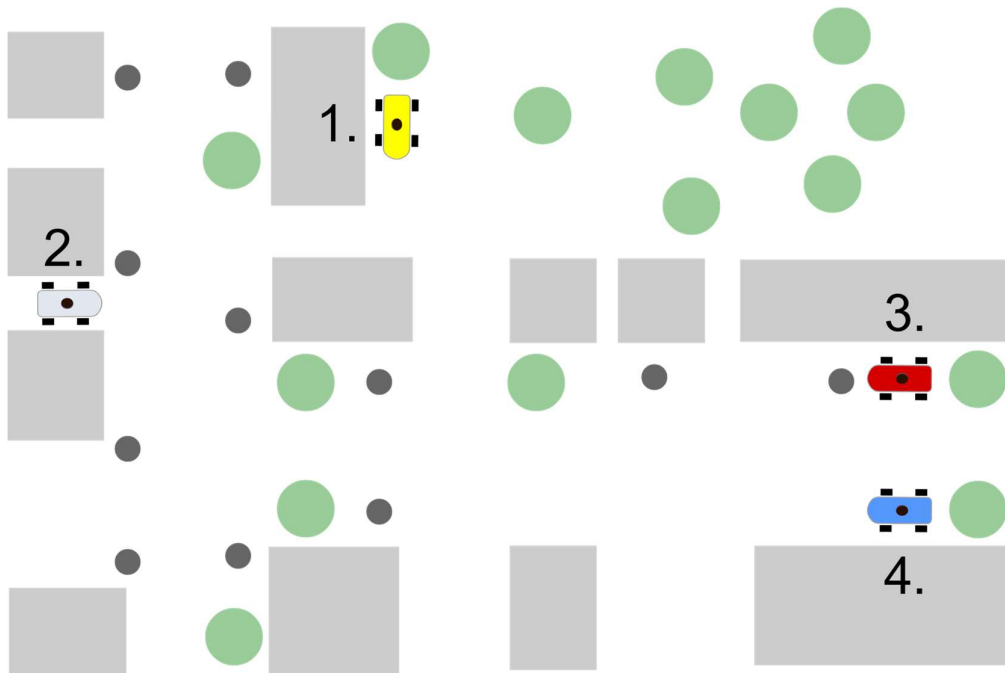
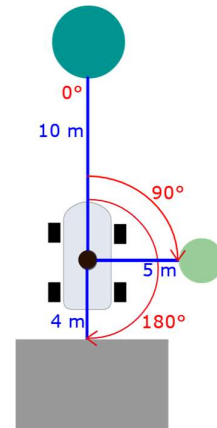
Röviddel azelőtt, hogy lemerült volna az akkumulátora, még sikerült egy parkolóhelyet találnia és modellt küldenie a környezetéről. Az elküldött modellben minden tárgyat, amit egy 360°-os érzékelővel elér, két értékkel ír le:

1. Az érzékelő szögével, mellyel a tárgy fele néz (lásd a képen). Az egyenes a 0°.
2. Az érzékelő tárgytól való távolságával.

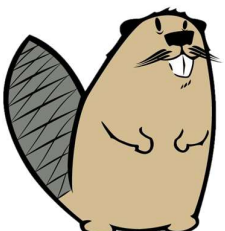
A példában például a modell:  $[(0, 10), (90, 5), (180, 4)]$

Az elveszett autó a következő modellt küldte haza:  $[(0, 5), (90, 4), (180, 5), (270, 12)]$

Találd meg az elveszett autót a térképen!



Melyik az elveszett autó?





### „A piros autó” a helyes válasz

Az elveszett autót a hazaküldött modell alapján a következő tárgyak veszik körül:

- az autóval szemben, 5m-re egy.
- a menetirány szerinti jobb oldalán, 4m-re egy.
- az autó mögött, 5m-re egy, és
- az autó bal oldalán, 12m-re egy.

A sárga autó nem lehet az elveszett autó, mivel a mögötte lévő tárgy (zöld kör) sokkal közelebb van, mint az előtte lévő (szürke téglalap).

A szürke autó sem lehet az elveszett autó, mivel a jobb és a bal oldalán lévő tárgyak közel egyforma (kis) távolságra vannak.

A kék autó sem lehet az elveszett, mert az előtte lévő tárgy (kis sötétszürke kör) sokkal távolabb van, mint a mögötte lévő (zöld kör).

A piros autó az elveszett autó, mivel az előtte és a mögötte lévő tárgyak közel ugyanakkora távolságra vannak és a jobb oldalán látható tárgy (szürke téglalap) sokkal közelebb van, mint a bal oldalon lévő (kék autó).

### MIÉRT INFORMATIKA?

Az önvezető (autonóm) járművek az úgynevezett LIDAR (light detection and ranging, magyarul: lézer alapú távérzékelés)-technológiát használják arra, hogy feltérképezzék a környezetüket. A navigációs alkalmazás minden, akár több száz méter távolságra lévő tárgyról (objektumról) egy összetett 3 dimenziós modellt hoz létre. A valósággal ellentétben a mi feladatunkban a modell nagyon egyszerű és csak a legközelebbi lévő tárgyakkal foglalkozik.

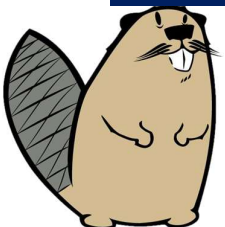
Általában egy modell a valóság egy leegyszerűsített leképezése, ahol csak azokat a szempontokat vesszük figyelembe, melyek valamilyen meghatározott célból fontosak számunkra. Az önvezető járművek esetében a LIDAR-modellt használják az összeütközések elkerülésére. Ebből a szempontból egyedül a tárgyak járműhöz viszonyított helye (pozíció) fontos. Az egyéb szempontok, mint pl. a tárgyak színe figyelmen kívül hagyhatóak.

### KULCSSZAVAK

Autonóm autó, lidar, modell

### WEBOLDALAK

<https://hu.wikipedia.org/wiki/LIDAR>





**JELSZÓ BIZTONSÁG (2018-DE-04)**

BENJAMIN – KÖZEPES

KADÉT – KÖNNYŰ

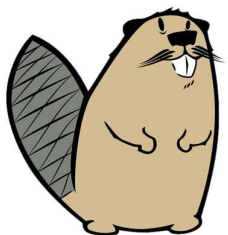
Karesz a következő három szabályt alkalmazza, amikor jelszót választ:

1. A jelszó legalább 8 karakter hosszú legyen.
2. A jelszóban szerepeljen legalább egy nagybetű is.
3. A jelszóban szerepeljen legalább egy szám (0 ... 9) vagy valamilyen speciális karakter (!, #, %, &, \*, +, <, >, ?)



Vajon melyik jelszó felel meg Karesz szabályainak?

- A. m1n2cr8fT
- B. t?ps!cr!t
- C. #tgnpvltm
- D. no20181112





**„A” válasz a helyes**

A helyes válasz az A. 8 karakter hosszú és van benne nagybetű és szám is.

B, C és D válaszok nem megfelelőek, mert nincs bennük nagybetű. A hosszuk rendben van, és a harmadik feltételnek is megfelelnek.

**MIÉRT INFORMATIKA?**

Sok személyes információt, adatot tárolunk a számítógépünkön vagy valamilyen felhasználói fiókhoz köthető tárhelyen. De a számítógépünk, hardver eszközeink is veszélyben lehetnek. Ezeknek védelemre van szükségük.

A legelterjedtebb alap védelem a jelszó használata. Melyet csak az az egy ember ismer, akié a jelszó.

De mitől biztonságos egy jelszó? Rengeteg jelszó-törő alkalmazás létezik, melyek többsége az úgynevezett „brute force” (nyers erő) módszeren alapul, amikor minden karakterkombinációt kipróbálunk, vagy pl. egy szótár szavait használjuk alapnak. Tehát semmiképpen sem jó, ha egy létező szót használunk, vagy rövid jelszót adunk meg.

Az amerikai NIST (National Institute of Standards and Technology) egy ebben a hód-feladatban leírt módszerhez hasonló ajánlást fogalmazott meg. Persze egy embernek nehéz egy nem túl rövid, értelmetlen karakterekből álló sorozatot megjegyezni. Ha jól megnézed a megoldásainkat, te is felfedezheted az „értelmét” a szavainknak. 😊

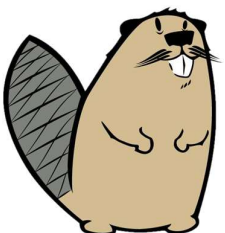
**KULCSSZAVAK**

Jelszó, biztonság

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Brute\\_force-támadás](https://hu.wikipedia.org/wiki/Brute_force-támadás)

[https://hu.wikipedia.org/wiki/National\\_Institute\\_of\\_Standards\\_and\\_Technology](https://hu.wikipedia.org/wiki/National_Institute_of_Standards_and_Technology)





### A HAZÚGOT HAMAR UTOLÉRIK (2018-HR-05)

BENJAMIN – NEHÉZ

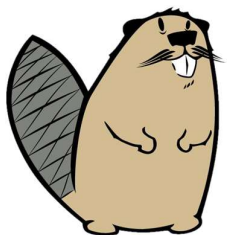
KADÉT – KÖZEPES

Egy szép napon Marci, Ilona, Dávid és Maja Annáék háza mellett focizik. Egyszer csak betörik az egyik ablak, és Anna szeretné tudni, ki törte be. Anna ismeri a gyerekeket és tudja, hogy hárman közülük mindig igazat mondanak. De a negyedik gyereknél nem tudja.

A négy gyerek a következőket állítja:



Vajon melyik gyerek törte be az ablakot?





**„Dávid” a helyes válasz**

Maja és Dávid állításai nem lehetnek egyszerre igazak, egyikőjük füllent. Ha Maja mondana igazat, akkor Dávid hazudna és Marci és Ilona állítása ugyanúgy igaz lenne.

Ha Dávid mondana igazat, Maja hazudna és akkor Marci és Ilona állítása sem lehetne igaz. De tudjuk, hogy hárman közülük igazat mondanak.

**MIÉRT INFORMATIKA?**

A feladat megoldásához logikusan kell gondolkodnod. Az ehhez alapul szolgáló logikát 1854-ben George Bool (1815-1864) fogalmazta meg, a logikai kifejezéseket alapelemeire vezette vissza és írta le.

Egy logikai kifejezés vagy igaz, vagy hamis (tertium non datur – kizárt harmadik elv). A kifejezéseket műveletekkel (operátorok) kombinálhatjuk. Az egyszerű logikai műveletek az ÉS (AND), a VAGY (OR) két kifejezésből hoznak létre egyet. De van olyan művelet (mint a NEM (NOT)), amelyik egy kifejezést vesz és azt változtatja meg.

Hogy a kombinált kifejezések mikor igazak, abban segíthet az úgynevezett igazságtábla.

Az egyik művelet a „HA”  $\rightarrow$  „AKKOR” következtetés, azaz implikáció. Ilyenkor beszélünk arról, hogy „logikus következtetést vonunk le”. Ez a művelet szükséges a feladat megoldásához.

A számítógépek alapját is ezek a Boole-kifejezések és egyszerű műveletek jelentik, mivel elég egyszerűek és egészen naggyá építhetőek.

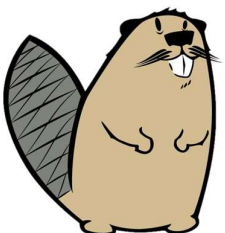
Vannak azonban számítógépek, melyek alapját más rendszer képezi. Például az 1950-es évek végén a három-értékű logikára (Lukasiewicz lengyel tudós ötlete) épült angol kezdeményezésű és orosz megvalósítású számítógép (ternary computer).

**KULCSSZAVAK**

Logika, kizárt harmadik

**WEBOLDALAK**

[https://hu.wikipedia.org/wiki/Kizárt\\_harmadik\\_elve](https://hu.wikipedia.org/wiki/Kizárt_harmadik_elve)  
[https://hu.wikipedia.org/wiki/George\\_Boole](https://hu.wikipedia.org/wiki/George_Boole)  
<https://hu.wikipedia.org/wiki/Boole-algebra>  
<https://hu.wikipedia.org/wiki/Implikáció>  
[https://hu.wikipedia.org/wiki/Többértékű\\_logika](https://hu.wikipedia.org/wiki/Többértékű_logika)  
[http://www.hiradastechnika.hu/data/upload/file/1978/01/1978\\_01\\_03.PDF](http://www.hiradastechnika.hu/data/upload/file/1978/01/1978_01_03.PDF)





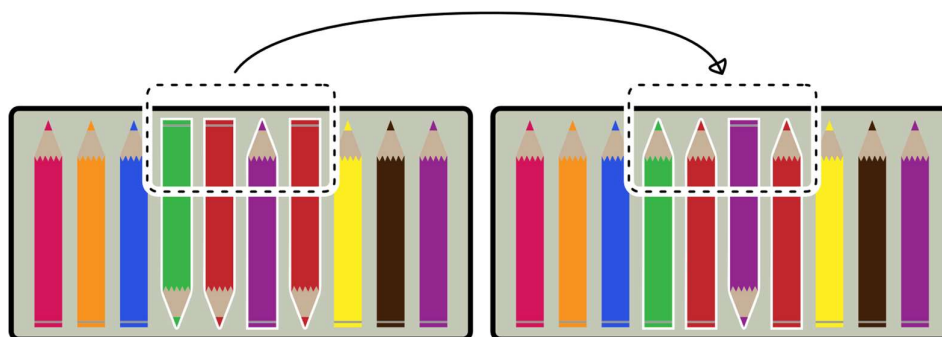
## ADA CERUZÁI (2018-HU-01)

KISHÓD – KÖZEPES

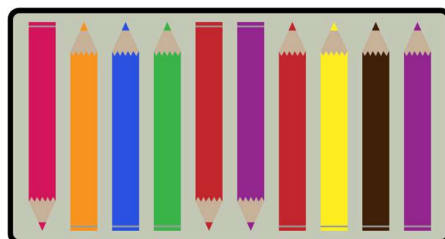
KADÉT – KÖNNYŰ

Adának van egy doboz színes ceruzája. Pár ceruza hegygel felfele és pár lefele van a dobozban. Ada azt szeretné, hogy mindegyik hegygel felfele nézzen.

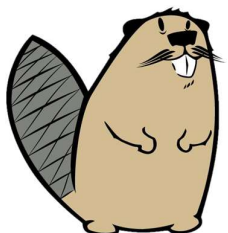
Egy fordítással egyszerre kettő vagy több, egymás mellett lévő ceruza irányát tudja megváltoztatni.



Most így állnak a dobozban a ceruzák:



Legkevesebb hány fordítással tudja Ada megfordítani a ceruzáit, hogy mindegyik hegygel felfele nézzen?

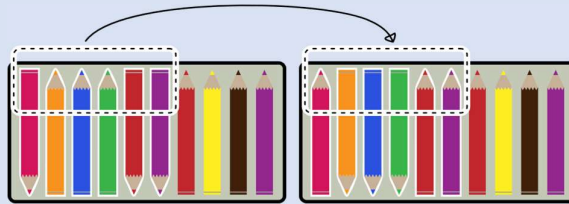




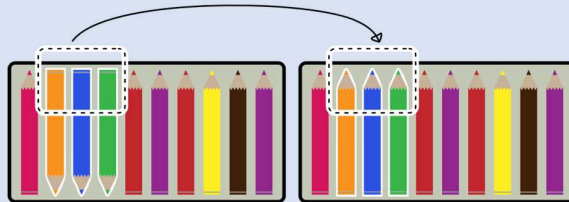
## „2” a helyes válasz

Elég kétszer 1-1 köteg, egymás mellett lévő ceruzát megfordítania.

Egy fordítás nem lenne elég, mert a lefele fordított ceruzák nincsenek egymás mellett. De ha Ada először az elsőtől a hatodik ceruzáig mindet megfordítja...



... azután a másodiktól a negyedekig lévőket visszafordítja, ...



Mindegyik ceruza hegygel felfele lesz a dobozban.

Természetesen az is egy két lépésből álló megoldás lenne, ha először megfordítja az első ceruzát, azután az ötödiket és a hatodikat együtt. De akkor nem lenne igaz, hogy kettő vagy több ceruzát kell egyszerre mozgatnia.

## MIÉRT INFORMATIKA?

Adának rengeteg ideje van, valószínűleg jóval több, mint ami két fordításra elég. És akár egyesével is forgathatja a ceruzáit. A számítógépek esetében ez nem ilyen egyszerű. Ha egy számítógépet programozunk, akkor mindig szabályokhoz kell tartanunk magunkat. Ráadásul nem ilyen kevés ceruzánál, de akár rengeteg adat feldolgozása esetén is.

Az adatok memóriakezelésénél: írásánál és olvasásánál a gyorsaság fontos. Általában az ilyen tárolásokat blokkok segítségével végzik. A számítógép egy blokkot tud egyszerre kiolvasni vagy írni. Ezzel időt tudunk spórolni. Egyébként minél nagyobb blokkokkal tudunk dolgozni, annál több időt spórolhatunk – de annál több (ideiglenes tárolás) helyet is fogunk használni. Általában ez mindig visszaköszön az informatika területén: vagy időt vagy helyet tudunk spórolni. De a kettőt együtt nem 😞

A feladat alapját az érdekes programozási feladatokat tartalmazó *Codechef* oldalról, Alei Reyes munkája képezi.  
<https://www.codechef.com/problems/ADACRA>

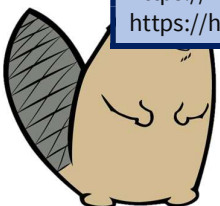
## KULCSSZAVAK

Hatékonyság, memória

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Tartós\\_állapotú\\_meghajtó](https://hu.wikipedia.org/wiki/Tartós_állapotú_meghajtó)

[https://hu.wikipedia.org/wiki/Memória\\_\(számítástechnika\)#Blokk\\_vagy\\_fájl\\_hozzáférésű\\_memória](https://hu.wikipedia.org/wiki/Memória_(számítástechnika)#Blokk_vagy_fájl_hozzáférésű_memória)





## VÍZESÉS (2018-HU-03)

BENJAMIN – NEHÉZ

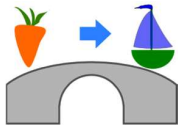
KADÉT – KÖZEPES

JUNIOR – KÖNNYŰ

Kata a hegyekben kirándul, ahol három vízesés van. A vízesések után a folyók egy idő után összefolynak egy tóba.

Kata egy répát vagy egy kishajót ejt a vízesések egyikébe.

A folyókon hidak ívelnek át, melyeket trollok őriznek. A trollok a hidak alatt kicserélnek bizonyos dolgokat, amik a folyóban sodródhatnak.

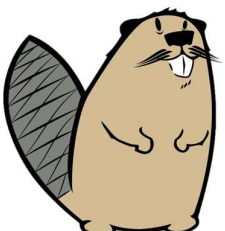
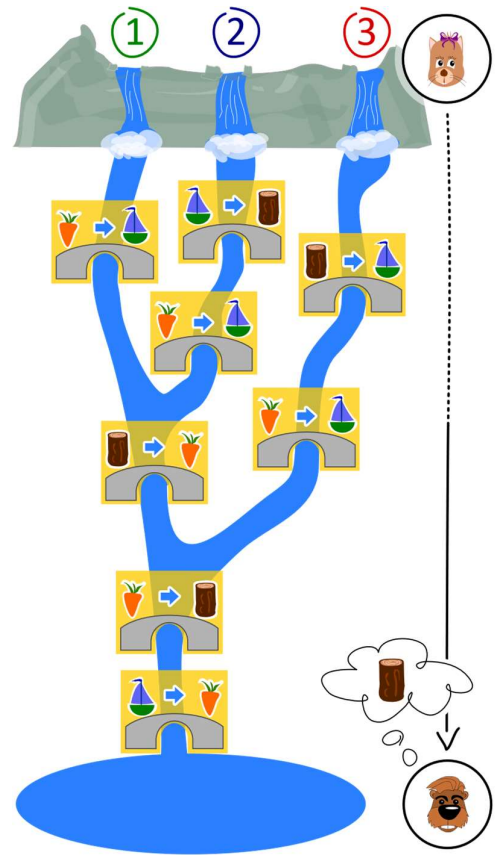


Például, ha egy répa egy ilyen híd alatt sodródik át, kicserélik azt egy kishajóra.

János a tónál áll.

Mit dobjon Kata és melyik vízesésbe, hogy János egy farönköt kapjon?

- A. Egy kishajót  az 1. vízesésbe.
- B. Egy kishajót  a 2. vízesésbe.
- C. Egy répát  a 2. vízesésbe.
- D. Egy répát  a 3. vízesésbe.





## „B” válasz a helyes

Mi történik az egyes válaszlehetőségek esetében?

A) Ha egy kishajót dobunk az 1. vízesésbe, akkor az csak az utolsó hídnál változik. Így Jánoshoz egy répa jutna el.

B) A 2. vízesésbe bedobott kishajót először répára, majd a répát egy farönkre cserélik a trollok. EZ tehát jó megoldás lehet.

C) Egy répát a 2. vízesésbe dobva, azt először kishajóra, majd a kishajót répára cserélik a trollok, így Jánoshoz répa jutna el.

D) A 3. vízesésbe bedobott répát először kishajóra, majd a kishajót ismét répára cserélik a trollok. Így Jánoshoz ismét csak egy répa jutna el.

Másik megoldási lehetőség lehet, ha visszafele kezdünk el gondolkodni.

Tudod, hogy a végén ahhoz, hogy farönköt kapj, a beejtett tárgy az utolsó előtti hídnál már csak répa lehet. Az egyetlen lehetőség, hogy ezen a ponton egy répa legyen az, ha az 1. vagy a 2. (de nem a 3.) vízesésből indulunk, mivel mindhárom vízesés utáni folyószakaszokon a répa minden esetben lecserélődik, de semmit sem cserélnek répára a trollok. Az 1. és a 2. vízesés folyószakaszainak összefolyásánál azonban egy fatörzset répára cserélnek! (Ezzel kiejtettük annak a lehetőségét, hogy az utolsó előtti hídhoz egy fatörzs érkezzon az 1. és 2. vízesésből. De az a lehetőség a 3. vízesés első hídjá miatt onnan sem lehetséges.)

Hogy fatörzs érjen az 1. és a 2. vízesés folyószakaszainak összefolyásánál álló hídhoz, az két módon történhetne meg:

- Egy fatörzset dobunk az 1. vízesésbe (ezt nem ajánlottuk fel a lehetőségek között)
- Egy kishajót dobunk a 2. vízesésbe (ez volt a B válaszlehetőség)

## MIÉRT INFORMATIKA?

A számítógépet úgy is elképzelheted, mint egy szerkezetet, mely bemeneteket olvas és kimeneteket ír.

Honnan tudja egy számítógép, hogy mit kell tennie (milyen bemenetre milyen kimenetet írjon)? A válasz az, hogy az emberek utasításokat adnak erre vonatkozóan.

Azaz programokat írunk, melyeket a számítógép végrehajt. Sok különböző programozási nyelv létezik. Egy programozó paradigma (általánosan elfogadott nézet) a funkcionális programozás. Ez a programozási stílus úgy működik, mint a számítógép maga – függvények (utasításcsoportok) meghívásából és kiértékeléséből áll.

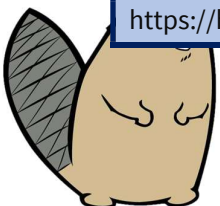
Ebben a hód-feladatban a hidak (a trollokkal) olyanok, mint egy-egy kis függvény, és az egész rendszer maga a program, amit funkcionális programnyelven írtunk meg.

## KULCSSZAVAK

Funkcionális programozás, programozási paradigma

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Programozási\\_paradigma](https://hu.wikipedia.org/wiki/Programozási_paradigma)  
[https://hu.wikipedia.org/wiki/Funkcionális\\_programozás](https://hu.wikipedia.org/wiki/Funkcionális_programozás)





## QUATTRIS (2018-HU-04)

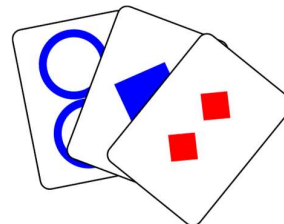
JUNIOR – NEHÉZ

SENIOR – KÖZEPES

A „Quattris” nevű kártyajátékban az egyes kártyalapokon alakzatok szerepelnek.

Az alakzatoknak 5 tulajdonsága lehet:

- A számuk (lehet 1 vagy 2)
- A színük (kék vagy piros)
- A formájuk (kör vagy rombusz)
- A forma nagysága (kicsi vagy nagy)
- A forma kitöltöttsége (teli vagy üres)

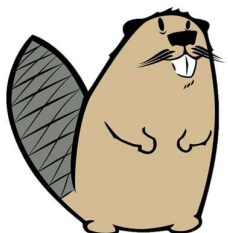


Minden lehetséges mintázat (tulajdonság-kombináció) pontosan egy kártyalapon szerepel.

A kártyák közül négy lap akkor alkot egy quattris-t, ha nincs olyan lapunk, amelyik valamely tulajdonságában egyedül van a 4 lap között.

Három kártyalapunk van az asztalon. Melyikkel egészíthetjük ki, hogy quattris-t kapjunk?

A.	B.	C.	D.





## „D” válasz a helyes

Mivel három kártyalapunk mindegyikén kicsi az alakzat, ezért olyan lapot nem választhatunk, ahol nagy. Így az A és a C kiesik.

Az alakzatnak pirosnak kell lennie, mivel csak egy piros alakzatunk van és az nem maradhat „egyedül”. Így a B-t is kizárhatjuk.

A D kártyalap tökéletes, mivel

- 2 alakzatot tartalmaz – így van 2 lapunk kettő és 2 lapunk egy alakzattal
- Rombusz – így van 2 lapunk, amin kör és 2 lapunk, amin rombusz szerepel.
- Üres alakzat – így van 2 lapunk, amin teli és 2 lapunk, amin üres alakzat szerepel.

## MIÉRT INFORMATIKA?

Csoportok létrehozása tulajdonságaik alapján igen gyakran előfordul az informatikában. Ebben a feladatban 5 tulajdonságot vizsgálunk 2-2 lehetséges értékkel és négy olyan kártyalapot kell beletennünk a csoportba, melyek közül egy sem lesz „egyedül” valamelyik tulajdonságára nézve.

Ezt halmazműveletekkel is modellezhetjük. (unió, metszet és különbség).

Hasonlóképpen dolgozhatunk adatbázisokban, ahol sokszor szükséges az adatokat közös tulajdonságaik alapján kiválogatni (leszűrni).

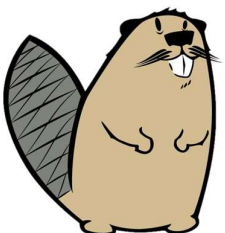
A feladat ötletét a valamivel bonyolultabb Set játék adta. Ott négy tulajdonság vehet fel 3-3 értéket és akkor kapunk set-et, ha az egyes tulajdonságok mindegyik kártyán egyformák, vagy mindegyik kártyán különbözőek.

## KULCSSZAVAK

Adatbázis, metszet, halmaz, tulajdonság

## WEBOLDALAK

<https://hu.wikipedia.org/wiki/Halmazelmélet>  
<https://hu.wikipedia.org/wiki/Adatbázis>





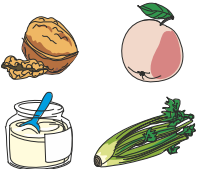
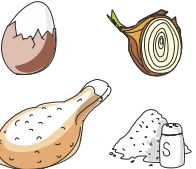



### HASONLÓ ÉTELEK (2018-HU-05)

KISHÓD – KÖZEPES

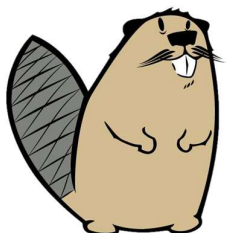
BENJAMIN – KÖNNYŰ

Egy szakács két ételt szeretne elkészíteni. Ezek az ételek nem lehetnek hasonlóak. A szakács akkor nevez két ételt hasonlóknak, ha legalább 2 összetevő ugyanaz a két ételben.

Spagetti	Tojássaláta	Waldorf-saláta	Tyúkhúsleves	Torta
				

Mely ételek hasonlóak?

- A. A tyúkhúsleves és a spagetti.
- B. A tyúkhúsleves és a Waldorf-saláta.
- C. A tyúkhúsleves és a tojássaláta.
- D. A Waldorf-saláta és a torta.





### „B” válasz a helyes

A helyes válasz a C): a tyúkhúsleves és a tojássaláta.

A tyúkhúslevesben és a tojássalátában is van tojás, hagyma és só.

A többi válaszlehetőségnél legfeljebb egy közös összetevő van. A tyúkhúsleves és a Waldorf saláta esetében nincs közös összetevő, a tyúkhúsleves és a spagetti esetében a hagyma, a Waldorf-saláta és a torta esetében szintén nincs közös összetevő.

### MIÉRT INFORMATIKA?

Sok esetben kell összehasonlítani dolgokat és kitalálni, hogy hasonlóak-e vagy sem. A biológusok hasonlítják pl. a baktériumok öröklődését, a kémikusok az anyagok tulajdonságait, a csillagászok a galaxist, a csillagokat és bolygókat és így tovább.

Dolgok összehasonlításához meg kell határoznunk, milyen tulajdonságokat szeretnénk vizsgálni. Ezekhez fogalmazhatjuk azután meg, mit jelent, hogy két dolog hasonló vagy nem hasonló. Például kijelenthetjük, hogy egy asztal és egy szék is fából készült, tehát hasonlóak. De az asztalt nem arra találták ki, hogy ráüljenek, míg a széket nem arra, hogy levelet írjanak rajta (bár mindkettő elképzelhető). Ezek alapján nem hasonlóak. De azt is mondhatjuk, hogy két szék csak akkor hasonló, ha azonos fából készült.

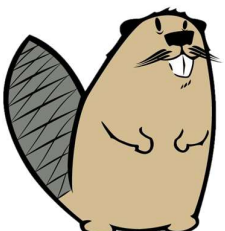
Ebben a hód feladatban öt étel szerepel 4-4 összetevővel. A biológusok, kémikusok, csillagászok és más tudományos területen dolgozók nem csak ilyen kevés számú dolgot hasonlítanak össze, hanem ezer, millió vagy milliárd dolgot, akár még több különböző tulajdonsággal, melyek mind szerepet játszhatnak a „hasonlóság”-ban. Itt jönnek a képbe az informatikusok, akik lehetővé teszik előre meghatározott hasonlósági paraméterek alapján a nagy adathalmazok automatizált hasonlítását.

### KULCSSZAVAK

Big data, hasonlóságelemzés, objektum, tulajdonság

### WEBOLDALAK

[https://hu.wikipedia.org/wiki/Big\\_data](https://hu.wikipedia.org/wiki/Big_data)



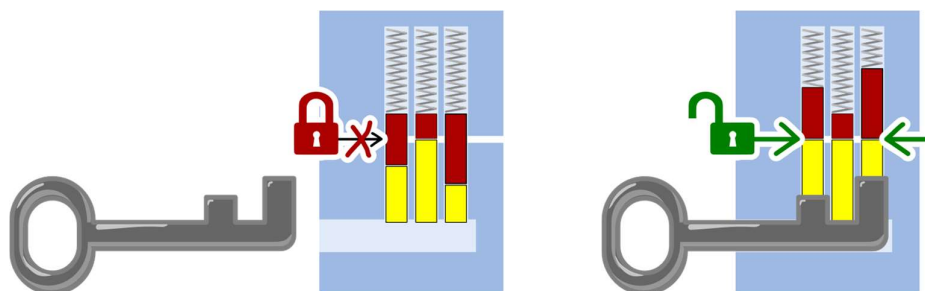


## BIZTONSÁGI ZÁR (2018-HU-06)

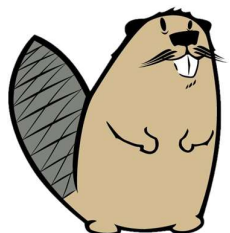
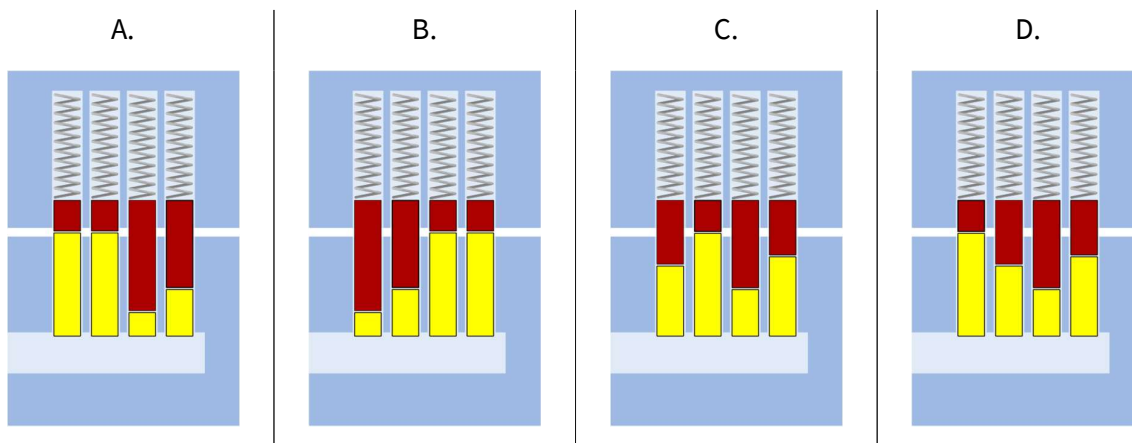
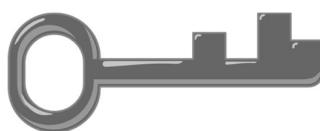
KISHÓD – NEHÉZ

BENJAMIN – KÖNNYŰ

Hugó egy zárkészítő műhelyben dolgozik, Az ott készült zárok a következőképpen működnek:



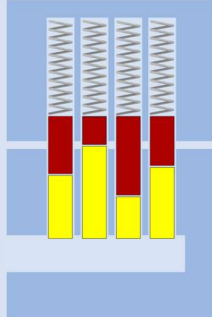
Melyik zárhoz tartozik ez a kulcs?





**„C” válasz a helyes**

Amikor a kulcsot betoljuk a zárba, a négy toll és a négy pöcök magassága együtt egyforma lesz, hogy a cylinder elfordítható legyen.

**MIÉRT INFORMATIKA?**

Ha egy összetett problémát részekre bontunk, sokszor találhatunk mintákat a kisebb problémákban.

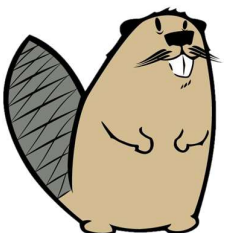
Egy megadott minta keresése és felismerése alapfeladat (a négy alap építőkö egyike) az informatika területén. Például egy szó előfordulása egy szövegben vagy hasonló képek keresése. Sokszor ha megtaláljuk a hasonlóságokat, mintát a felbontott, apróbb részekben, az segít hatékonyabban megoldani az összetett problémát.

**KULCSSZAVAK**

Mintafelismerés

**WEBOLDALAK**

<https://hu.wikipedia.org/wiki/...>





### LÁMPÁK ÉS KAPCSOLÓK (2018-HU-08)

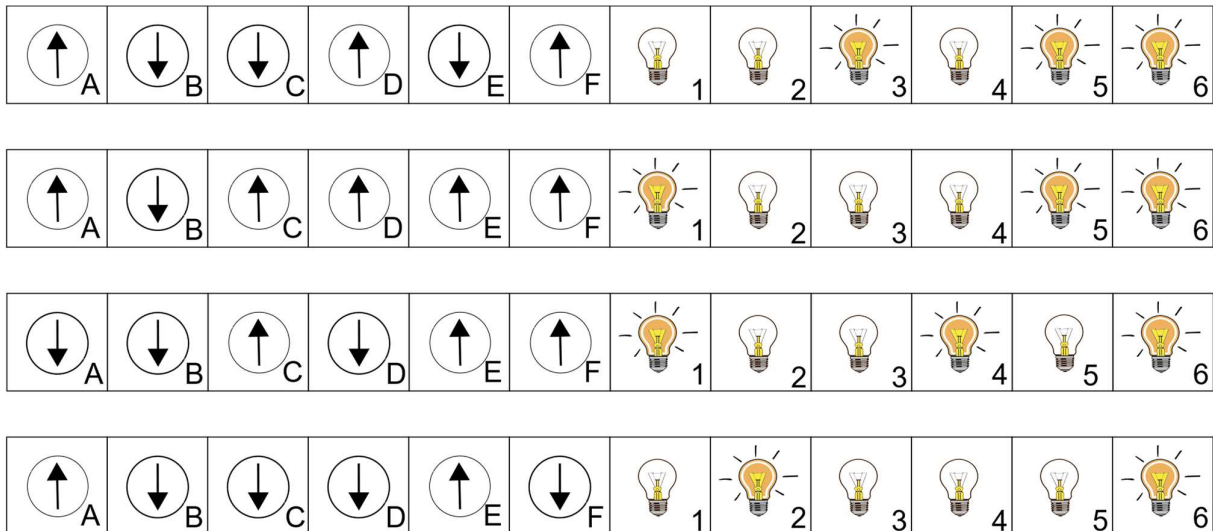
KADÉT – NEHÉZ

JUNIOR – KÖZEPES

SENIOR – KÖNNYŰ

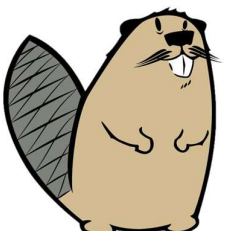
Tamás összekötött 6 lámpát 6 kapcsolóval. Minden kapcsoló pontosan egy lámpát kapcsol fel vagy le, de nem tudjuk, melyik kapcsoló melyik lámpához tartozik. A kapcsolók egy része akkor kapcsolja fel a hozzá kötött lámpát, amikor a nyíl felfele mutat, a többinél pedig fordítva.

Hogy megtaláljuk melyik kapcsoló, melyik lámpával van összekötve, a következő próbákat végeztük:



Melyik lámpát melyik kapcsoló kapcsolja?

- A. C-1, F-2, E-3, A-4, D-5, B-6
- B. C-1, E-2, D-3, A-4, F-5, B-6
- C. C-1, F-2, D-3, E-4, A-5, B-6
- D. C-1, F-2, B-3, A-4, D-5, E-6





### „A” válasz a helyes

Az első és a második próba esetében a C és az E kapcsolók és az 1. és a 3. lámpa különböznek. Ezért vagy C-1 és E-3 VAGY E-1 és C-3 lehetnek összekötve.

A harmadik és a negyedik próba esetében a többek között a C kapcsoló is különbözik, de a 3. lámpára ez nem volt hatással. Ezért a C-1 kell legyen a jó megoldás. Ebből következik, hogy E-3 a másik összekötés.

A harmadik és a negyedik próba esetében láthatjuk, hogy az A és az F kapcsolók a 2. és a 4. lámpákra hatnak. Tehát vagy A-2 és F-4 VAGY F-2 és A-4 a megoldás.

Az A kapcsoló azonban a második és a harmadik próba között megváltozott, de a 2. lámpa erre nem reagált. Ezért F-2 és A-4 a megoldás. Hasonlóan a D-5 kell legyen a kapcsolat, amiből automatikusan következik B-6.

A következtetéseinket csak az egymás alatt álló próbák összehasonlításával végeztük. Ez könnyebben végig vezethető, de tetszőleges próbákat is összehasonlíthattunk volna.

### MIÉRT INFORMATIKA?

Egy rendszer tesztelése és ezáltal belső felépítésének „kitalálása” egy igen gyakori eljárás mód.

Ha valaki, mint mi a feladat során, egy rendszert csak annak viselkedésének megfigyelésén keresztül vizsgál és a belső felépítésére a külső megfigyelése alapján vonatkoztat, akkor fekete doboz tesztről (black box test) beszélünk.

Ennek ellentéte az úgynevezett fehér doboz tesztelés (white box test), amikor a belső felépítést is figyelembe vesszük.

Mindkét fajta eljárás fontos szerepet játszik a hibakeresés (debugging) és hibajavítás folyamatában.

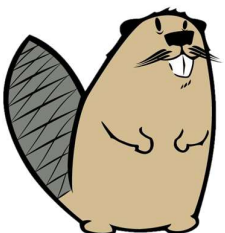
### KULCSSZAVAK

Fekete doboz, fehér doboz, tesztelés

### WEBOLDALAK

[https://hu.wikipedia.org/wiki/Feketedoboz\\_\(rendszerelmélet\)](https://hu.wikipedia.org/wiki/Feketedoboz_(rendszerelmélet))

[https://hu.wikipedia.org/wiki/Fehér\\_doboz](https://hu.wikipedia.org/wiki/Fehér_doboz)





### VÉGTELEN FAGYLALTÖLCSÉR (2018-IE-01)

BENJAMIN – KÖZEPES

Az utcában két fagyizó van. A kínálatukban ugyanaz a négyféle fagyalt található meg:

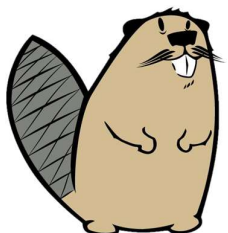
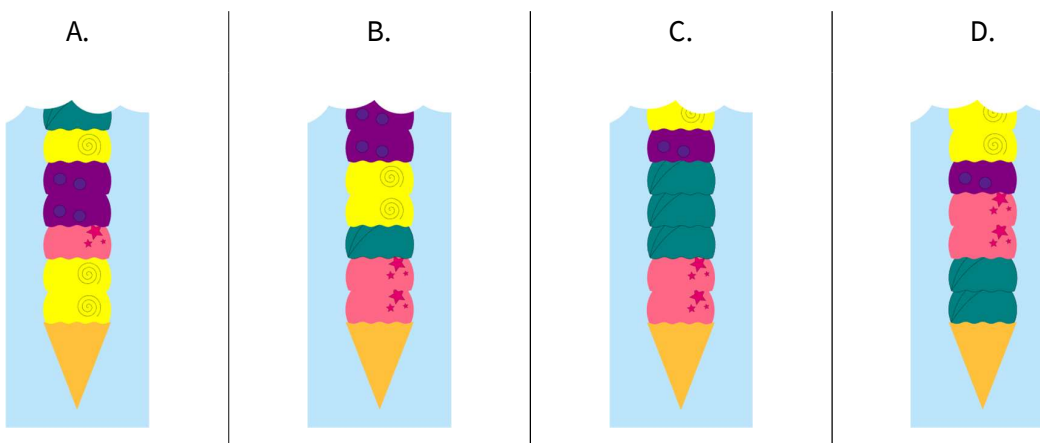


Az első fagyizó a következő útmutató szerint ad fagyit:

0. Vegyél egy üres tölcsért!
1. Válassz ki egy tetszőleges fagyit és vedd ki 2 gombócot ebből.
2. Vegyél egy gombócot valamelyik másik, az elsőtől különböző fagyiból.
3. Ha elég ennyi fagy, hagyd abba az adagolást. Különben az 1. lépéstől folytatd ismét.






A második fagyizó semmilyen útmutatást nem használ. A képen láthatod az alsó gombócokat.

**Melyik tölcsér származik biztosan a második fagyizóból?**








### „D” válasz a helyes

Ez az egyetlen olyan tölcsér, melyet nem tudunk összeállítani az első fagyizó útmutatása alapján. Helyesen 2 azonos gombóccal (  ,  ) kezdődik, melyet egy másik fajta gombóc (  ) követ. De ezután 2 különböző gombóc (  ,  ) következik, pedig az utasítás szerint 2 ugyanolyannak kellene lennie. Az A, B és C válaszban található tölcsérek legalább addig megfelelnek az útmutatásnak, amíg láthatóak.

### MIÉRT INFORMATIKA?

Valamilyen utasítást követve tölcsérekben, szövegekben vagy képeken is létrehozhatunk mintákat. A programozók kódjainak, programjainak mintái vagy mintától való eltérései is felismerhetők.

Időnként minták keletkeznek az utasítások ismétléseiből. Pl.  egy egyszerű minta, mely a  ,  ismétléséből jött létre. Ezt a mintát könnyű felismerni. A feladatban valamivel nehezebb, mivel az első fagyizó útmutatása véletlen döntéseket is tartalmaz.

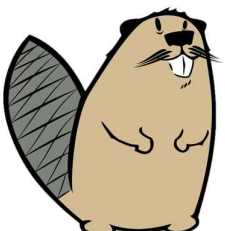
Alapjában véve sosem lehetünk biztosak abban, hogy egy sorozat véletlenül vagy utasítások során keletkezett-e. A feladat válaszlehetőségeinél csak egy tölcsérről tudtuk megmondani, hogy biztosan nem felel meg az útmutatásnak, és ezért biztosan a második fagyizóból származik. De azt egyik tölcsérről sem mondhatjuk biztosan, hogy az első fagyizóból származik.

### KULCSSZAVAK

Mintafelismerés

### WEBOLDALAK

<https://>





## SZIGORÚAN TITKOS (2018-IT-04)

### SENIOR – NEHÉZ

Xaver, Yvonne és Zoé rendszeresen lottóznak. Most tudódott ki, hogy valaki a városból nyerte meg a főnyereményt. Szívesen megtudnák, hogy közülük van-e a nyertes, de szeretnék titokban is tartani, ki az. Ezért a következőt teszik:

- Xaver és Yvonne feldob egy érmét.
- Xaver és Zoé feldob egy érmét.
- Yvonne és Zoé feldob egy érmét.

Minden esetben csak a 2 érintett nézi meg az egyes érmedobások eredményét.

Mindenki elmondja, hogy a két érme-feldobás eredménye „azonos” vagy „különböző”:

Annak, aki nem nyerte meg a lottó főnyereményt, igazat kell válaszolnia.

Ha valaki a lottó nyertese, akkor az eredmény ellenkezőjét kell válaszolnia.

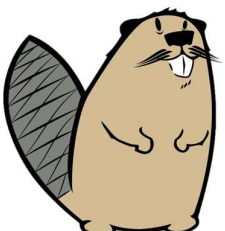
A képen egy példa látható az elvégzett érmedobásokkal és azzal a kitételrel, hogy Zoé nyerte meg a lottó főnyereményt. Nézd meg a következő helyzetet, ahol az érmedobás eredménye számodra nem ismert:



### A következő kijelentések közül melyik igaz?



- A barátok egyike sem nyert a lottón.
- A barátok egyike nyert a lottón, de nem tudjuk melyikük.
- A barátok egyike nyert a lottón, és pontosan tudjuk, ki.
- Nem tudjuk, hogy valaki nyert-e a lottón.





**„B” válasz a helyes** a barátok egyike nyert a lottón, de nem tudjuk, ki.

A feladat többféleképpen is megoldható. Egyet megmutatunk.

Három érme feldobásánál pontosan a következő két lehetőség állhat fenn:

- Mindhárom érme ugyanazt mutatja.
- Az egyik érme mást mutat, mint a másik kettő.

Ha senki sem nyert a lottón, a következő lesz érvényes:

- Ha minden érme egyforma, mindhárman „azonos”-at fognak mondani.
- Ha valamelyik érme mást mutat, akkor a barátok egyike „azonos”-t mond, a két másik pedig „különböző”-t.

Tehát kimondhatjuk, hogy minden más esetben a három barát egyike nyert a lottón. A mi különleges esetünkben a válaszok az „azonos”, „azonos”, „különböző”. Tehát elmondhatjuk, hogy valaki a három barát közül nem a valósat mondta.

## MIÉRT INFORMATIKA?

A Xaver, Yvonne és Zoe által használt módszer először az úgynevezett „étkező kriptográfusok protokolljában” (Dining cryptographers problem) került leírásra.

Ez a módszer azért különösen érdekes az informatikusok számára, mert mind a küldő, mind a fogadó névtelen marad a kommunikáció során: ha az eljárást helyesen vezetik le, a végén mindenki megbízhatóan tudja, hogy valamelyikük jelet adott (nyertes), de a jeladón (nyertes) kívül senki sem tudja ki volt az.

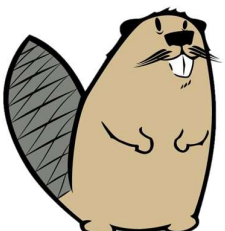
A fogadó is névtelen marad, mivel az összes résztvevő megkapja a jelet, üzenetet.

## KULCSSZAVAK

Kriptográfia, titkosítás, üzenet küldés

## WEBOLDALAK

<https://first.pet-portal.eu/?page=blog&topic=pet&func=read&id=72>





### HÓD KAPITÁNY NAPJA (2018-KR-08)

KADÉT – NEHÉZ

JUNIOR – KÖNNYŰ

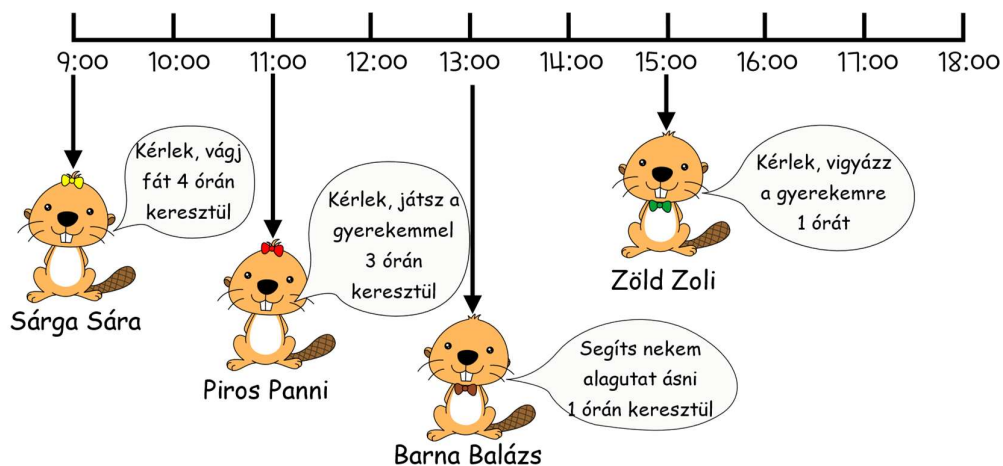
Hód kapitány szívesen segít, amikor egy másik hód megkéri valamire. Sajnos egyszerre csak egy feladatot tud megcsinálni. Amikor Hód kapitányt akkor kéri meg, hogy segítsen, amikor épp valami mást csinál, az alábbi fontossági táblázat alapján dönt:

Fontosság	Feladat
Magas	hód-babákról gondoskodni
Közepes	erdei munkák
Alacsony	minden más munka

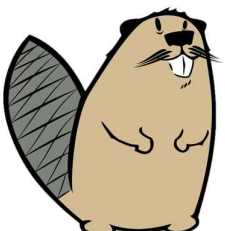
Amikor valaki segítséget kér Hód kapitánytól és az új feladat

- kevésbé fontos, mint az a feladat, amin épp dolgozik, nem szakítja meg a régi feladatot, és az új feladatnak várnia kell.
- ugyanolyan fontos, mint az a feladat, amin épp dolgozik, akkor óránként váltva dolgozik a feladatokon.
- fontosabb, mint az a feladat, amin épp dolgozik, akkor félbeszakítja a régi feladatot és végrehajtja az újat. Amikor azzal elkészül, akkor folytatja a félbehagyott feladatot.

Egy napon 4 hód kéri Hód kapitány segítségét, ahogy a képen látható:



Hány órákor fejezi be Hód kapitány Sárga Sára feladatát?





# HÓDÍTSD MEG A BITEKET – 2018-AS FELADATSOR

HÓD KAPITÁNY NAPJA (2018-KR-08)

## „17:00” a helyes válasz

Hód kapitány napirendjét az alábbi táblázatban láthatjátok:

	9:00 – 10:00	10:00 – 11:00	11:00 – 12:00	12:00 – 13:00	13:00 – 14:00	14:00 – 15:00	15:00 – 16:00	16:00 – 17:00	17:00 – 18:00
Sárga Sára									
Piros Panna									
Barna Balázs									
Zöld Zoli									

Sárga Sára közepes fontosságú feladatát 9:00-kor kezdi el Hód kapitány. De 11:00-kor félbeszakítja, hiszen Piros Panna feladata fontosabb. Piros Panna feladatát végrehajtja, Sárga Sára feladatának várnia kell. Barna Balázs feladata alacsony fontosságú, ezért nem szakítja félbe az éppen folyó feladatokat.

Miután Piros Panna feladatát befejezte, Hód kapitány összehasonlítja Sárga Sára és Barna Balázs feladatát. Mivel Sárga Sárának fontosabb, azzal folytatja a munkát. Egészen 15:00-ig, amikor Zöld Zoli magas fontosságú feladata miatt félbeszakítja. Egy óra múlva azt befejezi és ismét Sárga Sára feladatával folytatja (hiszen Barna Balázsé alacsony fontosságú). Mivel már csak 1 óra van hátra, ezért 17:00-kor végez ezzel a feladattal.

## MIÉRT INFORMATIKA?

Ez a hód-feladat olyan, mint a folyamat-ütemezés: egy operációs rendszeren belüli esemény, ahol több futó folyamat megosztja a számítási időt egy vagy több processzor között.

A folyamatok ütemezése lehetővé teszi, hogy az operációs rendszerek több feladatot látszólag párhuzamosan hajtsanak végre. Különböző folyamat-ütemezési eljárások léteznek, melyek például az átlagos feldolgozási időben vagy a maximális feldolgozási időben különböznek.

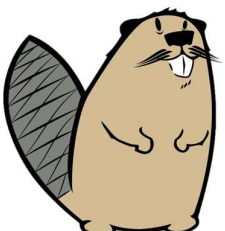
Ahogy feladat is mutatja, ilyen helyzet a mindennapi életben is előfordulhat, főleg akkor, amikor több szereplő kevés erőforrásért versenyez.

## KULCSSZAVAK

Feladat ütemezés, szál (processz)

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Szál\\_\(programozás\)](https://hu.wikipedia.org/wiki/Szál_(programozás))





## HÓD TÓ (2018-LT-02)

BENJAMIN – NEHÉZ

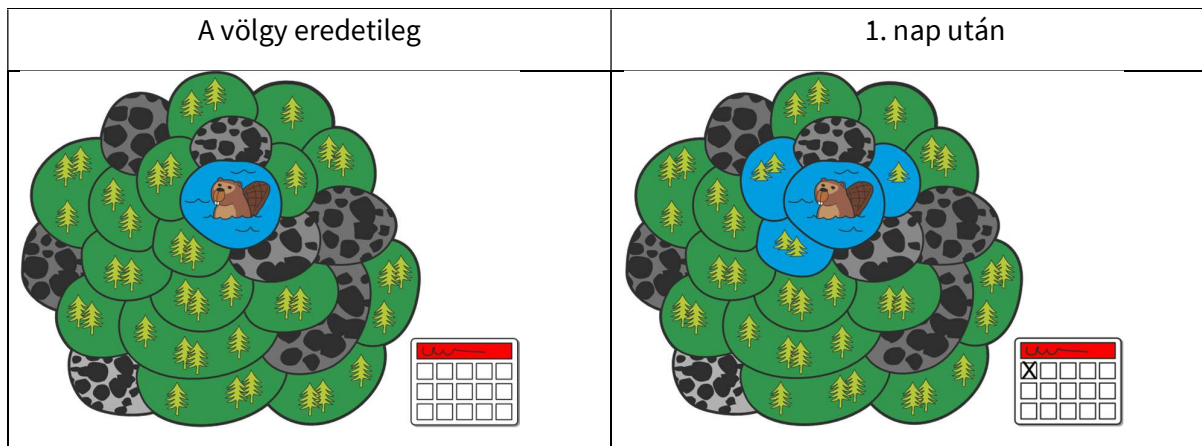
KADÉT – KÖZEPES

JUNIOR – KÖNNYŰ

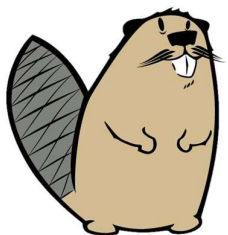
Egy völgyben, ahol hódok élnek van egy kis tó. Erdős és sziklás területek veszik körül.

Egy napon a hódok kinőtték a tavat és az erdős részeket szeretnék elárasztani. Minden nap elárasztják az összes olyan erdős területet, melyek a már elárasztott területeket határolják.

Az első napon három erdős területet árasztottak el.



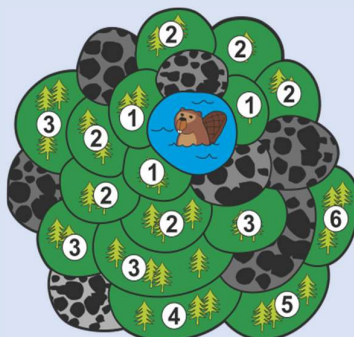
Hány nap után lesz elárasztva az összes erdős terület?





## „6 nap” a helyes válasz

A képen lévő számok megmutatják, melyik erdős területet, melyik napon tudták elárasztani a hódok.



Az 1-est tartalmazó területek közvetlenül a tavat határolják, így azokat az első napon, az ezekkel határos mezőket pedig a második napon tudták elárasztani. És így tovább... A 6. napon a völgy utolsó erdős területe is víz alá került.

## MIÉRT INFORMATIKA?

Ebben a hód feladatban a hódok egy összefüggő erdős területet árasztanak el, amely szomszédos rész-mezőkből – és az eredeti tóból – áll. A terület összefüggő, mivel minden erdős rész(terület) legalább egy másik erdős részterülettel szomszédos és minden erdős részterületről eljuthatunk bármelyik (és mindegyik másik) erdős részterületre erdős részterületeken keresztül.

Léteznek összefüggő területek a hód völgyön kívül is? Természetesen. Egy egyszínű terület egy (számítógép)képernyőn egy összefüggő, ugyanolyan színű képkockákból (pixelekből) álló terület. Fialatok egy olyan csoportja, ahol mindenki legalább egy emberrel barátságban van a csoporton belül is egy összefüggő terület, ha a barátságokat mint szomszédságot tekintjük.

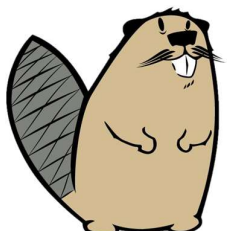
Az informatika ismer módszereket, amivel megállapíthatjuk, hogy egy terület összefüggő-e, mint például a szélességi vagy a mélységi keresés. Ezeknek a módszereknek a segítségével átszínezhettük egy kép egy részét vagy csoportot hozhatunk létre egy szociális hálón belül.

## KULCSSZAVAK

Gráf, szomszédsági gráf, gráf bejárás

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Összefüggőség\\_\(gráfelmélet\)](https://hu.wikipedia.org/wiki/Összefüggőség_(gráfelmélet))



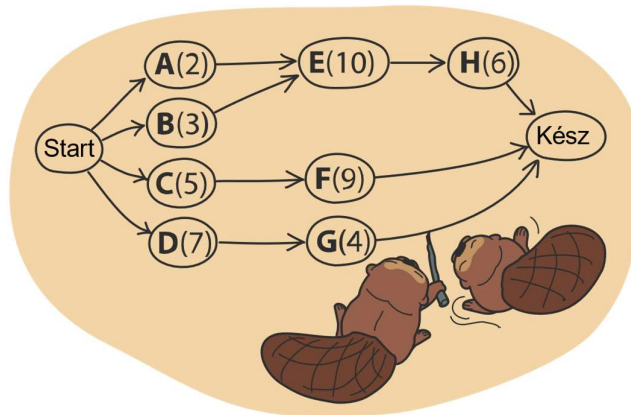


## KÉT HÓD DOLGOZIK (2018-LT-04)

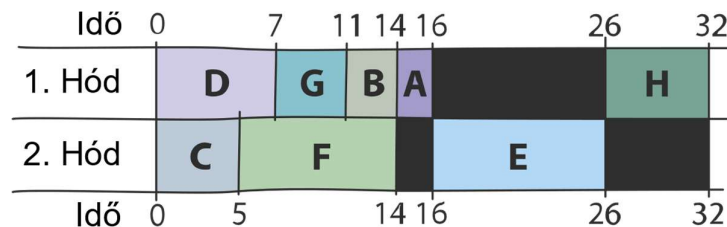
JUNIOR – KÖZEPES

SENIOR – KÖNNYŰ

Két hód egy gátat épít és ehhez 8 feladatot kell elvégezniük: kivágni a fákat, legallyazni azokat, a törzseket a vízhez vinni stb...

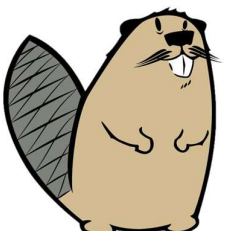


Minden feladatot egy betűvel és mögötte zárójelben egy számmal jelöltünk. A zárójelben található számok azt mutatják, hány óra szükséges a munka elvégzéséhez. Egy feladatot csak akkor tudnak elkezdni, ha bizonyos feladatokat már befejeztek. A sorrendet az ábrán a nyilak mutatják. A hódok együtt dolgoznak, de mindegyikük más-más feladaton. A következő ábra a hódok beosztását mutatja. A fekete rész a várakozással töltött időt jelzi.



Láthatod, hogy 32 óra alatt felépítik a gátat. De menne ez gyorsabban is!

**Mennyi az a legrövidebb idő, ami alatt a hódok felépíthetik a gátat?**





# HÓDÍTSD MEG A BITEKET – 2018-AS FELADATSOR

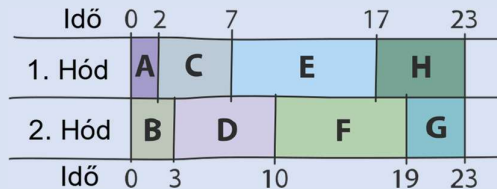
KÉT HÓD DOLGOZIK (2018-LT-04)

## „23 óra” a helyes válasz

Legalább 23 órára szükségük van.

A feladatban szereplő kép a két hód beosztását mutatja. Láthatod, hogy az első hód egy hosszú (10 órás) szünetet tart. A második hód pedig összesen 8 órát nem dolgozik. Gyorsabban készen lehetnének, ha ezt az időt is munkával tölténék.

Ha odafigyelünk arra, hogy a két leghosszabb feladatot (F és E) ne egy hód végezze, máris jobb tervet tudunk összeállítani. Például egy megoldás arra, hogy 23 óra alatt hogyan végezhetnek:



23 óránál kevesebb idő alatt nem tudnak végezni, hiszen így mindketten folyamatosan, szünet nélkül dolgoznak.

Ugyan össze tudnánk csoportosítani a feladatokat két 22 órás részre is ( $B(3)+E(10)+F(9)$  és  $A(2)+C(5)+D(7)+G(4)+H(6)$ ), de ekkor az egymásra való várakozásnál keletkeznie kellene plusz időnek (hiszen a 2. hód még nem készül el az F feladathoz szükséges feladatokkal).

## MIÉRT INFORMATIKA?

A feladatban szereplő tervhez a hódok a következő szabályt alkalmazták: „Válassz a még el nem kezdett feladatok közül egy olyat, ami a legtöbb időt veszi igénybe”.

Az informatikában ezt a stratégiát mohó algoritmusnak (angolul greedy-nek) nevezik. Először azt a részfeladatot oldjuk meg, amelyik ránézésre a lehető legnagyobb lépést teszi meg a megoldás felé. Sok esetben ez egy jó stratégia, de időnként – mint pl. ebben a feladatban – nem működik olyan jól.

Tulajdonképpen csak egy biztosabb út létezik ahhoz, hogy megtaláljuk a feladat legjobb megoldását: minden lehetséges tervet kipróbálunk, amelyik megfelel a megadott szabályoknak. Ezután eldöntjük, melyik a legjobb/leggyorsabb. Ez nagyobb projektek esetében olyan sok lehetséges megoldást jelent, hogy túl sok idő lenne mindet megvizsgálni és a döntést ilyen módon meghozni. Ezért hasznosak pl. a mohó algoritmushoz hasonló stratégiák akkor is, ha nem mindig a legjobb megoldáshoz vezetnek.

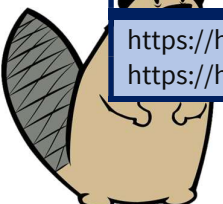
Ennek a feladatnak a problémáját direkt állítottuk úgy össze, hogy ne működjön a megoldáshoz a mohó algoritmus. Ilyen haszontalan feladat megfogalmazása igazi művészet. Az elméleti informatikában célzottan keresnek ilyen haszontalan feladatmegfogalmazásokat (angolul „worst case”), hogy az algoritmusok időigényét jobban tudják tesztelni.

## KULCSSZAVAK

Mohó algoritmus, időzítés,

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Mohó\\_algoritmus](https://hu.wikipedia.org/wiki/Mohó_algoritmus)  
[https://hu.wikipedia.org/wiki/Topologikus\\_sorrend](https://hu.wikipedia.org/wiki/Topologikus_sorrend)





**SOUNDEX (2018-PK-06)**

JUNIOR – NEHÉZ

SENIOR – NEHÉZ

Dénes szeretne szavakat a kiejtésük alapján kódolni. Ehhez a következőket teszi:

- Megtartja az első betűt (érintetlenül).
- Az összes többi betű közül kihúzza az A, E, I, O, U, H, W és Y betűket.
- A többi betűt számokkal helyettesíti a következőképpen:
  - B, F, P vagy V → 1
  - C, G, J, K, Q, S, X vagy Z → 2
  - D vagy T → 3
  - L → 4
  - M vagy N → 5
  - R → 6

Ha az előállított szóban egy szám többször szerepel, és a betűk, melyekből létrehozták ezeket, az eredeti szövegben közvetlenül egymás mellett álltak, csak egyszer tartja meg a jelölést. Ez akkor is érvényes, ha az első betű kerülne átalakításra.

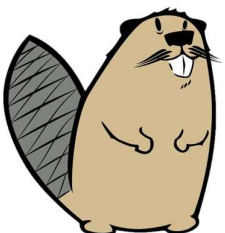
Végül csak az első 4 karaktert (beleértve az első megmaradt betűt) tartja meg. Ha nem lett 4 karakter hosszú az új szó, akkor hátulról 0-kal kiegészíti.

A Következő szavakat így kódolta át:

- |                    |                    |
|--------------------|--------------------|
| • Euler → E460     | • Kant → K530      |
| • Gauss → G200     | • Lloyd → L300     |
| • Heilbronn → H416 | • Lissajous → L222 |

**Melyik kódot állítja elő a Hilbert szóból?**

- A. H410
- B. B540
- C. H041
- D. H416





**„D” válasz a helyes**

Az első betű egy H, tehát az első karakter is ez lesz.

Ezután az összes A, E, I, O, U, H, W és Y betűt kihúzzuk, így a Hlbrt lesz az új szó.

A számcsereket követően a H4163 lesz az előállított szó.

Nincs egymás mellett ugyanolyan betűnk az eredeti szóban, így nem kell számot kitörölnünk.

Mivel csak az első 4 karaktert használjuk, így a megoldás a H416 lesz.

**MIÉRT INFORMATIKA?**

Ebben a hód-feladatban megismert eljárást soundex-nek (amerikai soundex) nevezik. Közel 100 éve Robert C. Russel és Margaret King Odell fejlesztette ki és szabadalmaztatta.

Ahhoz használták, hogy az angol nyelvben hasonló csengésű szavakat – neveket – találjanak. Ez azért működik, mert a betűk csoportja, melyek azonos kóddal (számmal) rendelkeznek, az angol nyelvben(!) hasonló csengésűek: a B, F, P és V ajakhangok, a C, G, J, K, Q, S, X és Z a szájpaddlásban képzett és sziszegő hangok, a D és a T foghangok, L egy hosszú folyékony réshang/laterális, az M és az N orrhangok és az R egy rövid folyékony réshang/laterális.

Mivel ez az eljárás elég egyszerű és az angol nyelv esetében meglehetősen jó eredményt ad, gyakran használják fonetikus, azaz hasonló csengésű szavak keresésénél. Sok adatbáziskezelő rendszerbe már alaptól beépítették.

A feladatban szereplő példák Donald Knuth-tól származnak, aki egyike a XX. század legnagyobb informatikusainak. Ő írta (illetve írja máig) a „Számítógép-programozás művészete” című könyvet, melynek 3. kötetében a „Rendezések és keresések”-ben található az eljárás leírása.

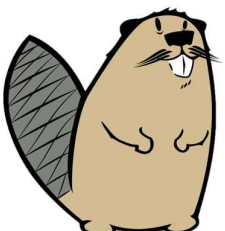
**KULCSSZAVAK**

Soundex, kódolás, fonetikus keresés

**WEBOLDALAK**

<https://en.wikipedia.org/wiki/Soundex>

[https://hu.wikipedia.org/wiki/Donald\\_Knuth](https://hu.wikipedia.org/wiki/Donald_Knuth)





## 29-ES HÁZIKÓ (2018-SK-07)

### KADÉT – NEHÉZ

Misi nyári munkát vállalt egy kempingben. A mai feladata, hogy a kemping faházaira feltegye a házszámokat.

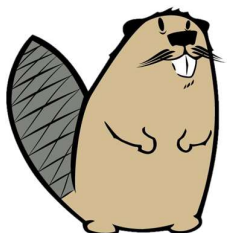
Egyes házakon már szerepelnek számok. A munkát az 50-es számnál kezdi és a 29-es tábla van a kezében. A következő utasításokat kapta a főnökétől:

- Balra menjen, ha a kirakandó szám kisebb, mint amelyik számú háznál éppen áll.
- Jobbra menjen, ha a kirakandó szám nagyobb, mint amelyik számú háznál éppen áll.
- Tegye fel a számtáblát a házra, ha egy olyan háznál áll, amelyik még nincs beszámozva.



Melyik házra kerül fel a 29-es szám, ha Misi pontosan követi az utasításokat?

- A. Az „I”-gyel jelölt házra.
- B. A „II”-vel jelölt házra.
- C. A „III”-mal jelölt házra.
- D. A „IV”-gyel jelölt házra.





**„A” válasz a helyes,** azaz a „I”-gyel jelölt házra.

A 29-es szám kisebb, mint az 50, ezért balra kell fordulnia. De a 29-es nagyobb, mint a 24-es, így annál a háznál jobbra fog fordulni. A 34-es ismét nagyobb, így ott ismét balra kell fordulnia. Az így elért háznak még nincs száma, ezért oda kell kitennie a 29-es számot.

## MIÉRT INFORMATIKA?

A házak számozása megfelel a bináris keresőfának, egy informatikában gyakran használt adatstruktúrának. Bináris keresőfa segítségével elmentett adatainkat gyorsabban/hatékonyabban találhatjuk meg.

Egy bináris keresőfa úgy épül fel, hogy minden elágazásnál (csomópontban) egy „elem” kerül eltárolásra. Minden elágazásból 2 út (él) vezethet további elágazásokhoz. Egy új elem eltárolásakor például mindig a bal oldali út kerül kiválasztásra, ha az új (tárolandó) elem értéke kisebb, mint az elágazásnál lévő elem értéke. Különben a jobb oldali úton megy tovább. Az új elem az első szabad (üres) kereszteződésbe kerül eltárolásra.

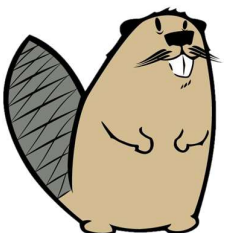
Mikor vissza akarunk keresni egy elemet, könnyen el tudjuk dönteni melyik utat válasszuk.

## KULCSSZAVAK

Bináris fa, keresés, algoritmus

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Fa\\_\(adatszerkezet\)#Bináris\\_fa](https://hu.wikipedia.org/wiki/Fa_(adatszerkezet)#Bináris_fa)





## ÜZENET A SULI MENEDZSMENT RENDSZERBŐL (2018-TR-02)

KISHÓD – NEHÉZ

BENJAMIN – KÖZEPES

KADÉT – KÖZEPES

JUNIOR – KÖNNYŰ

SENIOR – KÖNNYŰ

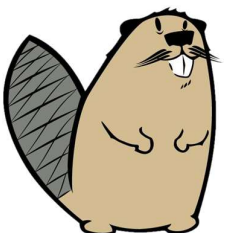
Kaptál egy üzenetet az Suli Menedzsment Rendszerből a következő tárggyal: „Frissítsd az adataidat!”.

Az üzenetben az áll, hogy küld el az azonosítódát (id) és a jelszavadat, különben a hozzáféréseid minden adatoddal együtt törlődni fog a rendszerből.

**Az alábbiak közül melyik lenne a legjobb lépés, melyet meg kellene tenned?**

1. Küld el az azonosítódát és a jelszavadat egy válaszlevélben.
2. Küld el egy barátod azonosítóját és jelszavát egy válaszlevélben.
3. Töröld ki a levelet válasz nélkül.
4. Küld el pár barátodnak az azonosítódát a jelszavaddal együtt.
5. Szólj az iskolában a tanárodnak és a szüleidnek a levélről.

- A. 1, 4.  
B. 2, 4.  
C. 2, 5.  
D. 3, 5.





#### „D” válasz a helyes

Ilyen esetben az a legjobb, ha nem válaszolsz a levélre és azonnal jelzed a környezetekben lévő felnőtteknek, hogy mivel találkozál.

#### MIÉRT INFORMATIKA?

Az adathalászat, vagy más néven phishing egyik formája az, amikor levélben általad ismert cégnek, szervezetnek a nevében próbálnak meg bizalmas adatokat begyűjteni tőled, például a jelszavadat egy adott rendszerhez.

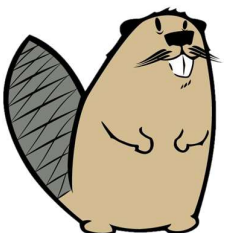
Ez ellen természetesen léteznek már szűrő-alkalmazások is, de fontos, hogy te is figyelj arra, hogyan kezeled a bizalmas adataidat (mint például a jelszavaid)

#### KULCSSZAVAK

Biztonság, adathalászat

#### WEBOLDALAK

<https://hu.wikipedia.org/wiki/Adathalászat>





## A FÖLDÖNKÍVÜLI (2018-TR-06)



BENJAMIN – KÖNNYŰ



Egy földönkívülinek van egy feje, egy teste, két karja és két lába. A következő utasítások alapján át tud alakulni úgy, hogy 1-1 testrésze akár többször is módosul:

Utasítások:

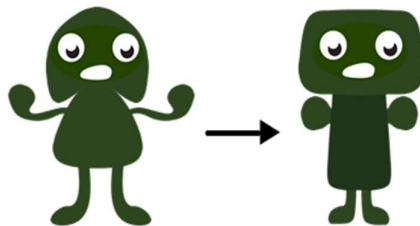
$F(k)$  : fej kerek  lesz,  $F(4)$  : fej négyszögletes  lesz,  $F(3)$  : fej háromszögszerű  lesz

$T(k)$  : test kerek  lesz,  $T(4)$  : test szögletes  lesz,  $T(3)$  : test háromszögletű  lesz

$K(+)$  : karok hosszúak  lesznek,  $K(-)$  : karok rövidek  lesznek

$L(+)$  : lábak hosszúak  lesznek,  $L(-)$  : lábak rövidek  lesznek.

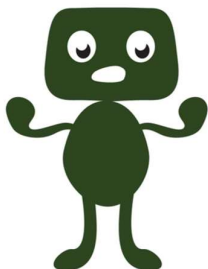
Például:  $F(k)$ ,  $T(4)$ ,  $F(4)$ ,  $K(-)$ ,  $L(-)$



Hogy néz majd ki egy földönkívüli, ha a következő utasításokat követi?

$F(3)$ ,  $L(+)$ ,  $T(3)$ ,  $K(+)$ ,  $F(k)$ ,  $K(-)$ ,  $T(k)$ .

A.



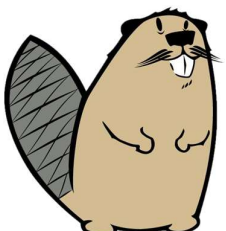
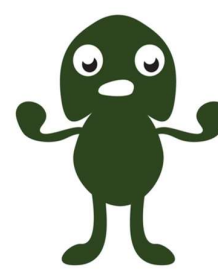
B.



C.



D.





## „B” válasz a helyes

Minden testrésze csak az utolsó utasítása vonatkozik. Korábbi utasítások ugyanerre a testrésze nem lesznek hatással, mivel a következő utasítások felülírják.

Ezért az eredmény egy olyan földönkívüli, akinek kerek a feje, kerek a teste, rövid a karja és hosszú a lába. Ez a B válasz.

## MIÉRT INFORMATIKA?

Egy program végrehajtása az utasítások sorrendjében történik. A később következő (és végrehajtott) utasítások így ismét meg tudják változtatni a korábbi utasítások hatásait. A számítógépes programokban rendszerint használhatunk változókat, melyekben értékeket tárolhatunk.

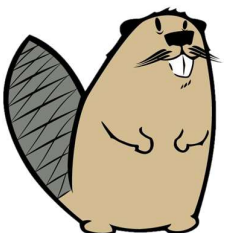
Egy testrész megváltoztatásához – ahogy ebben a hód-feladatban történik – használhatunk változókat. Pl. egy F nevű változóban tárolhatjuk el a fej formáját. Egy fejforma hozzárendelésén keresztül az F változó értékét változtatjuk meg.

## KULCSSZAVAK

Programozás, szekvencia

## WEBOLDALAK

<https://hu.wikipedia.org/wiki/Programozás>





ÚTHÁLÓZAT (2013-TW-02)

KISHÓD – NEHÉZ

BENJAMIN – KÖZEPES

KADÉT – KÖNNYŰ

A térképünkön négy metróvonal van, melyek a következő állomásokról indulnak: „Könyvtár”, „Pályaudvar”, „Kultúrház” és „Főtér”.

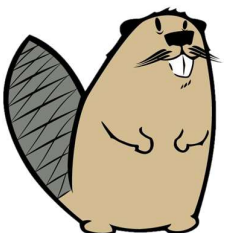
Négy olyan megálló van, ahol az egyik metróvonalról a másikra átszállhatunk: „Városház”, „Múzeum”, „Malom” és „Piac tér”.



Jancsi el szeretne jutni az Állatkertbe. Csak azt tudja, hogy egyszer kell átszállnia egy másik metróvonalra.

**Vajon melyik metróvonalon kezdi meg az utazását?**

- A. 4 süteményt.
- B. 5 süteményt.
- C. 6 süteményt.
- D. 7 süteményt.

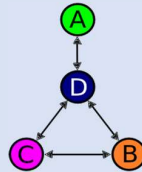




„D” válasz a helyes azaz Főtér – Kórház metróvonal.

Ha az Állatkerttől visszafele utazunk, csak egy olyan állomás („Városközpont”) van, ahol át tudunk szállni másik metróra. Ez a „Főtér” és a „Kórház” között közlekedik.

A metróvonal hálózatot egy gráf segítségével is felrajzolhatjuk: ahol azt jelöljük, hogy melyik metróvonalról melyikre szállhatunk át közvetlenül.



Melyikről	Melyikre		
Könyvtár ↔ Állatkert	Főtér ↔ Kórház		
Pályaudvar ↔ Tó	Kultúrház ↔ Park	Főtér ↔ Kórház	
Kultúrház ↔ Park	Pályaudvar ↔ Tó	Főtér ↔ Kórház	
Főtér ↔ Kórház	Pályaudvar ↔ Tó	Kultúrház ↔ Park	Könyvtár ↔ Állatkert

Tehát ha valaki egy átszállással szeretne a Könyvtár – Állatkert metróvonalon utazni, akkor csak a Főtér – Kórház vonalról szállhatott át.

## MIÉRT INFORMATIKA?

Ismerős neked a vonalhálózat térképe? Valóban, sok busz, villamos vagy metró (esetleg ezek közös) vonalhálózata hasonlóan néz ki. Ez nem véletlen, hanem Harry Beck találmánya, aki 1931-ben a londoni földalattihoz tervezett egy ilyen térképet.

Az informatikában az ilyen elvont (absztrakt) térképeket gráfoknak nevezzük, melyek csomópontokból (állomások) és élekből (vonalszakaszok az állomások között) állnak. A hód-feladatban szereplő térképen megkülönböztethetünk még köztes csomópontokat, ahonnan csak 1 vagy 2 él vezet ki (végállomások, normál állomások) és csomópontokat, melyeknek több éle van (kereszteződések, átszálló helyek).

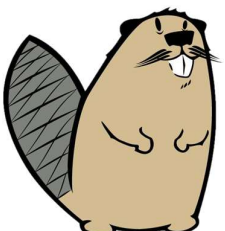
Gráfokat még több helyen használnak. Emberek szociális kapcsolatainak ábrázolására, útvonaltervezéshez vagy vásárlási ajánlatok/javaslatok elkészítésére. Tehát az is fontos informatikával kapcsolatos kompetencia, hogy tudj gráfokkal bánni 😊

## KULCSSZAVAK

Gráf, közlekedési hálózat, térkép

## WEBOLDALAK

<https://hu.wikipedia.org/wiki/Gráf>





**KIFESTŐ (2018-UK-01)**

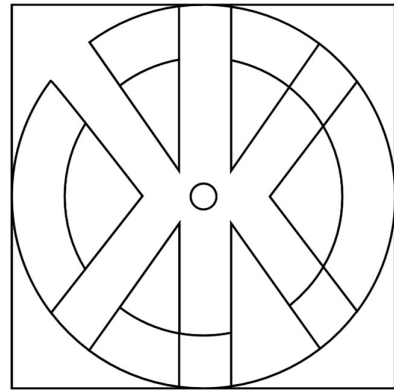
KISHÓD – KÖZEPES

BENJAMIN – KÖNNYŰ

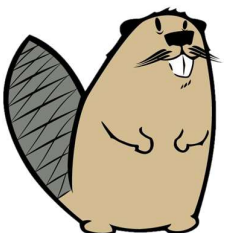
A hódok készítettek egy kifestőt. A színezéskor az összefüggő területeknek – ahova vonal átlépése nélkül átjutunk – ugyanaz lesz a színe.

Úgy szeretnék kiszínezni, hogy két szomszédos (egymás melletti, vonallal határolt) területnek különböző színe legyen. Nem számít szomszédos területnek az, ahol csak csücskök érnek össze.

Ehhez pedig a legkevesebb színt szeretnék felhasználni.



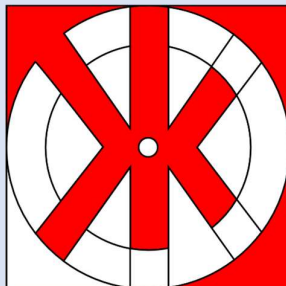
**Legkevesebb hány különböző színre lesz szükségük?**



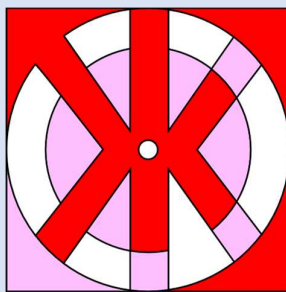


**„3 szín” a helyes válasz,** azaz 3 különböző színre lesz szükségük.

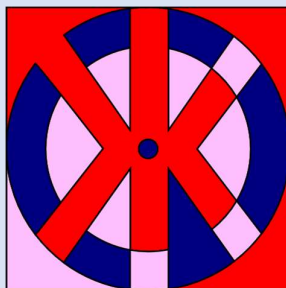
Kezdjük például egy piros színnel és színezzük ki a jobb felső sarkot és az összes olyan területet, ami ezzel nem szomszédos terület:



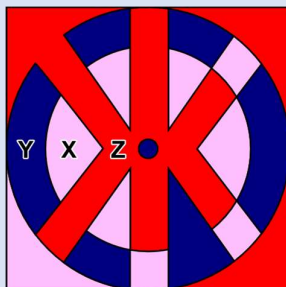
Ha egy második színnel (pl. halványlila) a bal alsó sarkot és az összes nem szomszédos és még be nem színezett területet beszínezzük, ezt kapjuk:



Tulajdonképpen készen is vagyunk, mivel az ábránk piros, halványlila és fehér színnel be van színezve a feltételnek megfelelően. De természetesen a fehér területeket is beszínezhetjük (pl. sötétkékkel).



Kevesebb, mint három szín nem lesz elég. Az X-szel jelölt terület balról az Y-nal jelölt területtel szomszédos, ezért különböző színűeknek kell lenniük. De mindkét terület szomszédos a Z-vel jelölt területtel. Ezért annak egy harmadik színre lesz szüksége.



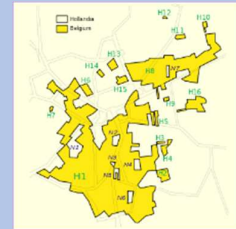


## MIÉRT INFORMATIKA?

Legkevesebb hány színre van szükségünk, hogy egy tetszőleges területet kiszínezzünk úgy, hogy két szomszédos terület nem lehet azonos színű?

A helyes válasz a 4.

Négy szín elég, ha nincs „beékel” területünk. Beékel terület (enklávé) például Büsingen am Hochrein vagy Campione d’Italia vagy egészen meglepően Baarle (<https://hu.wikipedia.org/wiki/Baarle-Nassau>) Hollandia és Belgium területén.



Ezt az elméletet 4-szín tételnek nevezzük.

Annak bizonyítása, hogy négy szín elég, nem könnyű. 200 évvel ezelőtt bizonyították azt, hogy 5 színnel biztosan megoldható a feladat. Csak 1976-ban tudta Kenneth Appel és Wolfgang Haken matematikus bizonyítani, hogy négy szín is elég. Ahhoz már számítógépet használtak, hogy kivételeket és ellenpéldákat vizsgáljanak, mivel kézzel nem lehetséges minden esetet megnézni. Manapság is sokszor használnak matematikusok számítógépet bizonyításokhoz.

A 4-szín tételt sok helyen használják: például repülési tervek összeállításánál, amikor repülési folyósókat kell felosztani, hogy a repülők mindig biztonságos távolságot tartsanak egymástól. Vagy akár frekvencia területek kiosztásakor, hogy azok ne zavarják egymást, de a vétel a sok adóantenna ellenére ne legyen rosszabb.

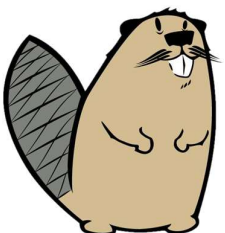
## KULCSSZAVAK

Négyszín tétel, színezés

## WEBOLDALAK

<https://hu.wikipedia.org/wiki/Négyszín-tétel>

[https://hu.wikipedia.org/wiki/Enklávés\\_exklávés](https://hu.wikipedia.org/wiki/Enklávés_exklávés)





NAPIREND (2018-UK-02)

JUNIOR – NEHÉZ

SENIOR – KÖZEPES

Négy hód szeretne esti iskolában tanulni a Hód Egyetemen. Öt különböző tanfolyam közül választhatnak:

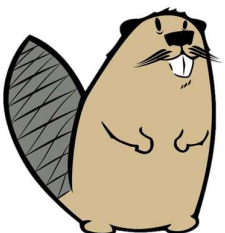
- Természettudomány (TT)
- Technológia (TC)
- Mérnöki tudományok (ME)
- Művészet (MŰ)
- Matematika (MA)

A hódok nagyon lelkesek és mindegyikük több tanfolyamot is szeretne választani:

- Alfonz ME, MŰ és TC tantárgyakat szeretné;
- Betti TT, TC és MŰ tantárgyakat;
- Cecil MA és MŰ tantárgyakat;
- Diana ME, MA és MŰ tantárgyakat.

A Hód Egyetem minden tanfolyamot egy héten csak egy alkalomra szeretne meghirdetni és a lehető legkevesebb estét szeretné beütemezni.

**Legalább hány estére szervezzenek tanfolyamokat, ha azt szeretnék, hogy mind az öt hód beiratkozhasson az általa választott tanfolyamokra?**



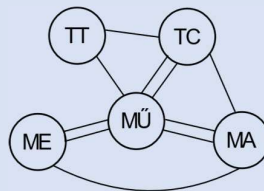


**„3 este” a helyes válasz,** azaz 3 estére kell a tanfolyamokat beosztaniuk.

Minden diák szeretne Művészetet (MŰ) tanulni, ezért az egyik estére minden diáknak ez a tanfolyamot kell beütemezni. Ha Alfonz igényeiből indulunk ki, még kell két este, amikor a Mérnöki tudományok (ME) és a Technológia (TC) lesz meghirdetve.

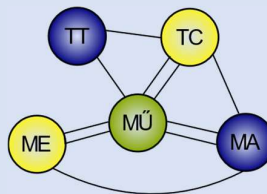
Mérnöki tudományokat (ME) Diana is szeretne tanulni, ezért a nála még hiányzó Matematikát (MA) beütemezhetik párhuzamosan (egy napra) a Technológiával (TC). Ez Cecilnek is megfelel. Betti Természettudomány (TT) tanfolyamát pedig meghirdethetik párhuzamosan (egy napra) a Mérnöki tudományokkal (ME).

Másik megoldási lehetőség, ha próbálkozás helyett megrajzolunk egy konfliktus-gráfot:



Itt a csomópontok lesznek a tanfolyamok. Két tanfolyamot összekötünk, ha egy hód mindkettőt szeretné.

Második lépésben színezzük ki a gráf csomópontjait úgy, hogy az összekötött csomópontok nem lehetnek azonos színűek. A kérdés már csak az, legfeljebb hány színnel tudjuk ezt megoldani. Annyi estére lesz szükség.



## MIÉRT INFORMATIKA?

Az ilyen kisebb, kevesebb összefüggésből álló feladatot könnyen megoldhatjuk okoskodással (azaz logikával), következtetésekkel is. De ha diákok százairól, több tanfolyamról és egyéb bejövő feltételekről (pl. valamelyik tanár szerdán nem ér rá) kell döntenünk, már igen nehezen boldogulnánk valamilyen stratégia, algoritmus nélkül.

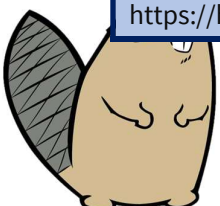
Az egyik, erre a problémára (is) megoldást nyújtó algoritmus a „gráfszínezési” algoritmus. Ez az informatikának egy igen aktívan kutatott területe a mai napig, mivel nagyon nagy számú csomópont esetében nem feltétlenül oldható meg elfogadható időn belül egyértelműen egy-egy színezés. A gráfszínezés gyakorlati alkalmazását megtalálhatod például a mintaillesztésnél, sportesemények lebonyolítási rendjének meghatározásánál, ülésrendeknél, vizsgák időbeosztásánál, taxik, kiszállítók menetrendjének összeállításánál, sudoku rejtvények megoldásánál

## KULCSSZAVAK

Gráf, gráfszínezés, algoritmus

## WEBOLDALAK

[https://hu.wikipedia.org/wiki/Gráfok\\_színezése](https://hu.wikipedia.org/wiki/Gráfok_színezése)









### HÁROM BARÁT (2018-VN-03)

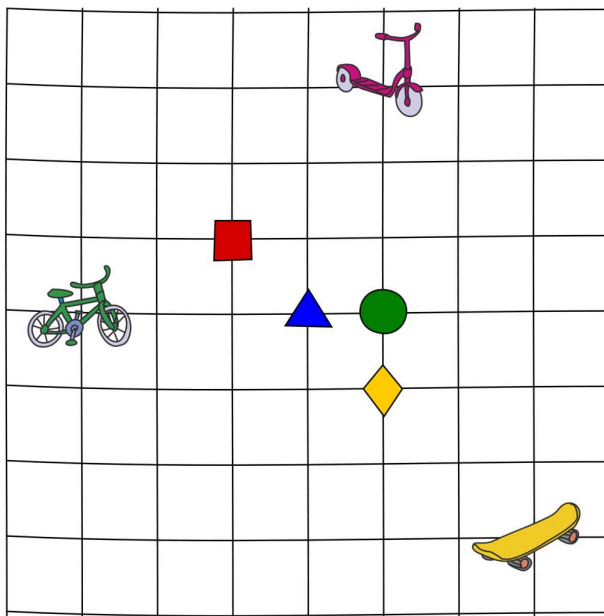
JUNIOR – KÖNNYŰ

A kereszteződések, ahol Béla , Anna  és Jutka  lakik, a térképen a járművükkel jelöltük.

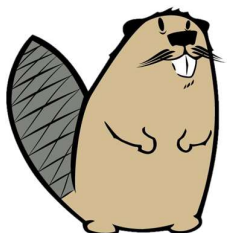
Mindhárman szeretnének egy olyan találkozási pontot megbeszélni, amely összességében a legrövidebb utat jelenti. Út hosszának a kereszteződéstől kereszteződésig tartó kis szakaszokat értik.

A , ,  és  szimbólumok a lehetséges találkozási pontokat jelölik.

Például Annától  a  találkozási ponthoz a legrövidebb út hossza 6.

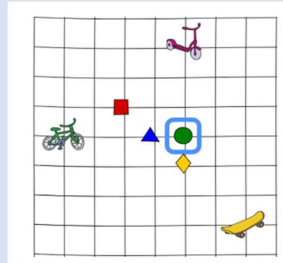






Melyik szimbólum jelöli a legjobb találkozási pontot a barátoknak?





„●” a helyes válasz



-  Nem helyes, mert a piros négyzethez vezető utak hosszának összege  $4 + 3 + 8 = 15$ .
-  Nem helyes, mert a kék háromszöghöz vezető utak hosszának összege  $4 + 3 + 6 = 13$ .
-  Helyes, mert a zöld körhöz vezető utak hosszának összege  $3 + 4 + 5 = 12$ , azaz a legkisebb.
-  Nem helyes, mert a sárga rombuszhoz vezető utak hosszának összege  $4 + 5 + 4 = 13$ .

### MIÉRT INFORMATIKA?

Ahhoz, hogy a találkozási pontok közül a legjobbat megtaláljuk, kiszámoljuk a barátok mindegyikének a távolságát az egyes pontoktól, majd ezeket összeadjuk és kiválasztjuk a legalacsonyabbat.

Ez elég egyszerű, és nem sok idő – akár fejben is megy. De több barát és több lehetséges találkozási pont esetében ebből optimalizálási feladat (probléma) válik, aminek a megoldása nem megvalósítható időt határoz meg.

Az informatikában vannak olyan eljárások, melyek ilyen optimalizálási problémára megvalósítható időn belül csak egy megközelítőleg optimális megoldást találnak.

Egy ezek közül a lokális keresés (local search): ahhoz, hogy a barátok nagy számára egy megközelítőleg optimális megoldást (találkozási pontot) találjunk, a lokális keresésnél kijelölünk egy elsődleges megoldást. Ez történhet véletlenszerűen is. Az ehhez a ponthoz és az ezzel szomszédos találkozási pontokhoz tartozó út-hosszak összegét hasonlítjuk össze, és a legjobbat kiválasztjuk.

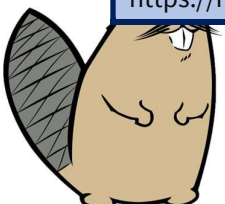
A következő lépésként innen vizsgálódunk tovább. Egészen addig, amíg egy elfogadható megoldást nem találunk (vagy elfogy az időnk, nem tudunk továbblépni, ...).

### KULCSSZAVAK

Keresés, optimalizálás, lokális keresés

### WEBOLDALAK

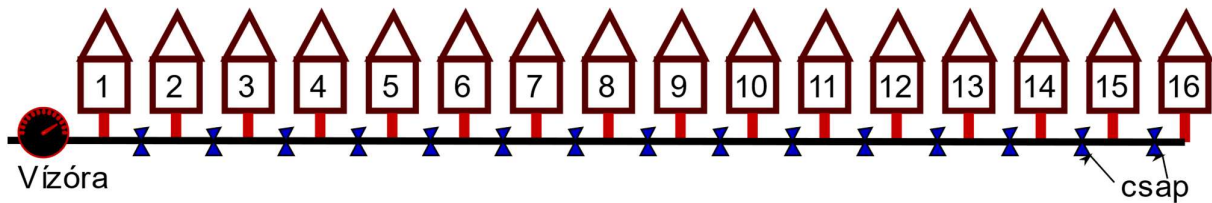
<https://hu.wikipedia.org/wiki/Keresőalgoritmus>





### TALÁLD MEG A REPEDÉST (2018-ZA-01)

KADÉT – KÖNNYŰ



A 16 ház vízvezetékének egyike megrepedt, így ott kifolyik a víz. Meg kell találni, melyik háznál van a repedés. Hogy megkönnyítsük a keresést, egyik házban sem használnak vizet a keresés idején.

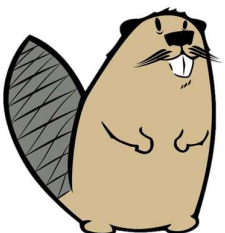
A vizsgálatot a következőképpen folytatjuk le:

A gyanús házakhoz tartozó csapot elzárjuk és megnézzük a vízórát, hogy még mindig folyik-e a víz. Majd újra kinyitjuk a csapot.

Ha pl. kezdetben a 8-as és a 9-es ház közötti csapot zárjuk el, és a vízóra még mindig a víz folyását (fogyasztást) jelzi, akkor tudjuk, hogy a repedés az 1-es és a 8-as házak közt van valahol (és nincs repedés a 9-től 16-ig számozott házaknál).

Tudjuk, hogy csak egyetlen repedés keletkezett a házaknál.

**Mennyi a legkevesebb vizsgálat, mellyel biztosan megállapíthatjuk a repedés helyét függetlenül attól, hol található a repedés?**





### „4” a helyes válasz

A legkedvezőbb (optimális) stratégia lényege, hogy a szóba jöhető házak között „félúton” zárjuk el a csapatot. Kezdetben 16, majd 8, 4, 2 és végül csak 1 házat jelent.

Legelőször zárjuk el a csapat közepén, azaz a 8-as és a 9-es ház között. Ha továbbra is folyik a víz, akkor a résnek az 1-től 8-ig számozott házak egyikénél kell lennie. Ha már nem folyik, akkor a 9-16 házak egyikénél. Ezután zárjuk el a középső csapat az érintett házak között (azaz az első esetben a 4-es és az 5-ös, a második esetben a 12-es és a 13-as ház között). Így megint felezni tudtuk a gyanús házak számát.

Pontosan 4 ilyen lépés után biztosan megtaláljuk azt a házat, ahol a repedés keletkezett.

Ha nem mindig a felénél választunk csapatot, és a nagyobb (több házat lefedő) részben van a repedés, akkor nem lesz elég a 4 lépés.

### MIÉRT INFORMATIKA?

Az a keresési stratégia, amikor a szóba jöhető elemeket minden lépésben felezzük, a bináris keresés. Csak akkor működhet, ha az elemek nem csak úgy vannak rendezve, hogy kijelölhetjük a „közepüket”, de az is megállapítható, hogy a keresett elem melyik félben található.

Például egy könyvárban egy könyv akkor kereshető így, ha a könyvek cím szerint sorba vannak rendezve, és így megállapíthatjuk, hogy melyik felében található a keresett könyv.

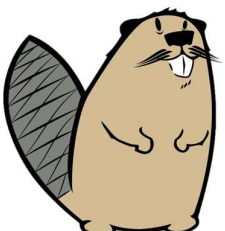
A bináris keresés esetében a vizsgálatok száma valóban jóval kevesebb, mint ha úgy keresnénk, hogy minden elemet sorban megvizsgálunk. 1000 könyv esetében a bináris kereséssel csak 10 keresési lépést szükséges végrehajtanunk.

### KULCSSZAVAK

Keresés, bináris keresés, algoritmus

### WEBOLDALAK

<https://hu.wikipedia.org/wiki/Keresőalgoritmus>





### TÁMOGATÓINK, KÖSZÖNETNYÍLVÁNÍTÁS

Köszönjük a nemzetközi Bebras kezdeményezés országainak, kiemelten a DACH-csapatnak  
(Németország, Ausztria, Svájc),  
az ELTE IK hallgatóinak, illetve  
a kapcsolattartó tanároknak szervezői munkájukat.

A HÓD VERSENY MINDEN TARTALMÁRA A CC BY-NC-SA 4.0 LICENSZ VONATKOZIK.

