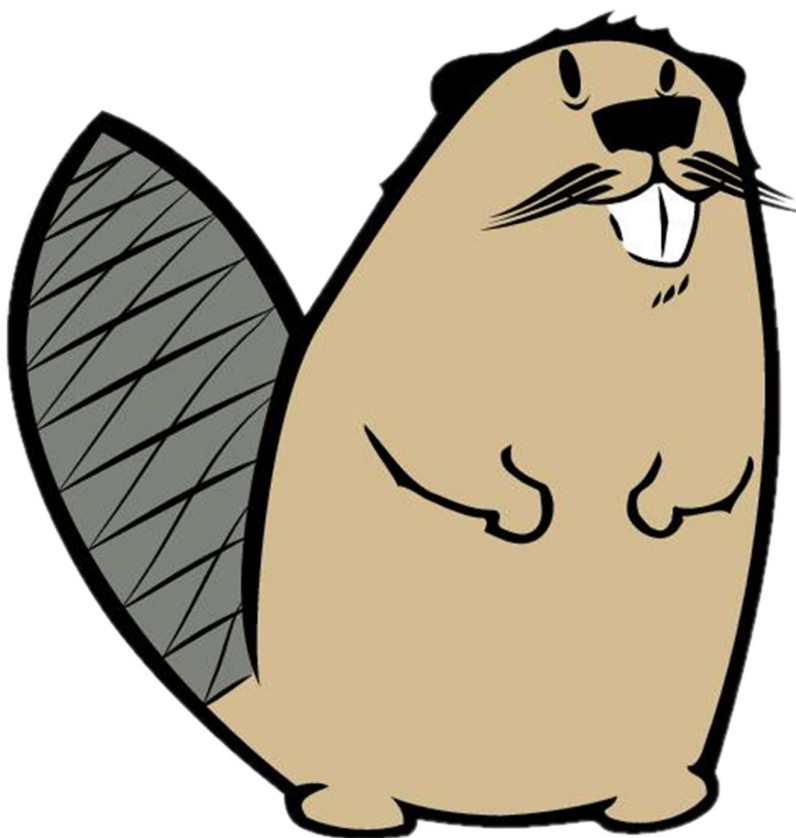


HÓDÍTSD MEG A BITEKET!

**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 megválaszolásukhoz. 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.

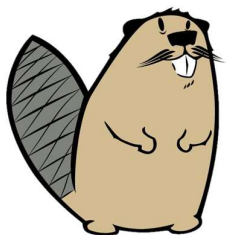
További információkért látogasson el a [HTTP://E-HOD.ELTE.HU/](http://E-HOD.ELTE.HU/) weboldalra, vagy írjon email-t az info@e-hod.elte.hu címre.

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

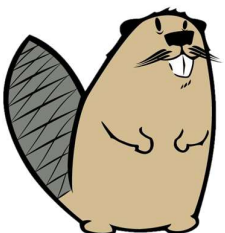
A Kishód korcsoportban 11, minden más korcsoportban 18 feladatot kell megoldani három nehézségi szinten. Minden helyes válasz pontot ér, minden helytelen válaszért pontlevonás jár.

Nem megválaszolt kérdés esetében az összpontszám változatlan marad.

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

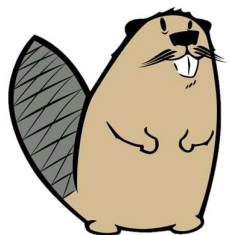
	Könnyű	Közepes	Nehéz
Helyes válasz	6 pont	9 pont	12 pont
Helytelen válasz	-2 pont	-3 pont	-4 pont

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
Sütemény elvitelre (2013-AT-11)	5
Át a sötét barlangon (2016-CH-04a)	7
Mosolyt kérek! (2017-DE-02).....	9
Szomszédok (2013-AT-11).....	11
Számítógépes játék (2018-CH-03).....	13
Barátlátogatás (2018-CH-05)	15
Felvonó a Matterhorn-on (2018-CH-07)	17
Z bolygó (2018-DE-02).....	19
Az elveszett autó (2018-DE-03).....	21
Jelszó biztonság (2018-DE-04)	23
A hazúgot hamar utoléri (2018-HR-05)	25
Ada ceruzái (2018-HU-01).....	27
Vízesés (2018-HU-03)	29
Lámpák és kapcsolók (2018-HU-08)	31
Hód kapitány napja (2018-KR-08).....	33
Hód tó (2018-LT-02)	35
29-es házikó (2018-SK-07)	37
Üzenet a Suli Menedzsment RendSzerből (2018-TR-02)	39
Úthálózat (2013-TW-02).....	41
Találd meg a repedést (2018-ZA-01).....	43
Támogatóink, köszönetnyilvánítás	45

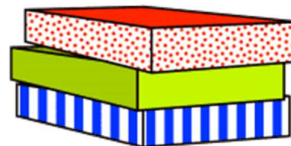


SÜTEMÉNY ELVITELRE (2013-AT-11)

KADÉT – KÖZEPES

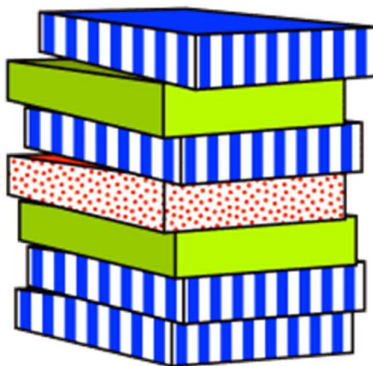
Tibi cukrász, és egyszerre mindig három süteményt süt.

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

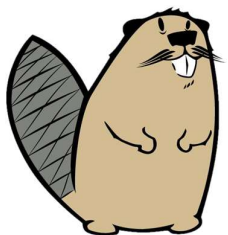


A hármas kupacot azonnal Tomihoz, az eladóhoz viszi. Tomi a hármas kupacot az eladásra szánt dobozok tetejére teszi. Ha Tomi elad egy süteményt, mindig a legfelső csomagot veszi le az eladásra szánt dobozok közül.

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 hármashalomból sem hiányzik egy sütemény sem. Az is lehet, hogy Tomi nyolc vagy tizenegy süteményt is eladott már, hiszen ha egy hármashalomból 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.

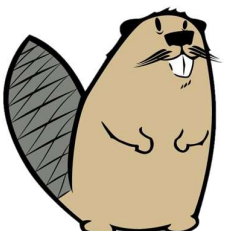
A verem koncepciót az informatika gyakran alkalmazza, például alprogramok behívásánál. A számítógép egy veremben jelöli, hol kell a programnak továbbfutnia, ha az alprogram elkészült. A verem tehát praktikus, mert így az alprogram még egy másik al-alprogramot is képes behívni és így tovább...

KULCSSZAVAK

Verem, Adatszerkezetek

WEBOLDALAK

http://hu.wikipedia.org/wiki/Verem_%28adatszerkezet%29



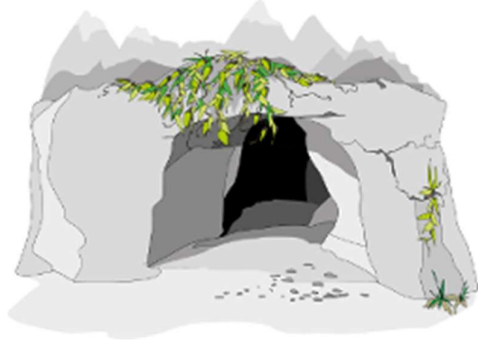
ÁT A SÖTÉT BARLANGON (2016-CH-04A)

KADÉT – NEHÉZ

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.

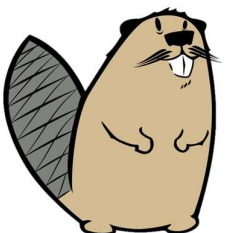
A keskeny és sötét barlangon biztonságosan csak egy vagy két hód mászhat át egy időben. Mivel csak egy zseblámpájuk van, több fordulót kell tenniük. Amikor ketten mennek, mindig annyi időre van szükségük, amennyire a lassabbnak.

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.



HÓDÍTSD MEG A BITEKET – 2018-AS FELADATSOR

ÁT A SÖTÉT BARLANGON (2016-CH-04A)

„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.

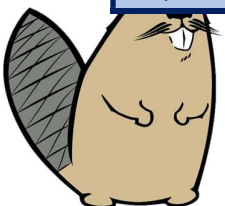
Hasonló elgondolások vezették az író, Isaac Asimovot 1942-ben, az első számítógépek készítésekor. Ő vezette be, többek között ezt a törvényt: „A robotnak nem szabad kárt okoznia emberi lényben vagy tétlenül tőnnie, hogy emberi lény bármilyen kárt szenvedjen.” Mivel az önjáró autó is egy robot, szintén 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

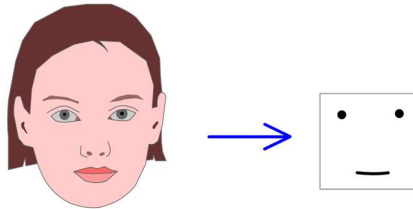
https://hu.wikipedia.org/wiki/Matematikai_logika



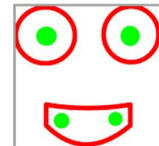
MOSOLYT KÉREK! (2017-DE-02)

KADÉT – KÖNNYŰ

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

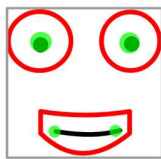


Ezt követően az alkalmazás egy zöld pontokból és piros vonalakkból álló sablon segítségével megvizsgálja a modellt.

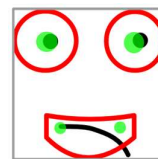


Mosolygós arcként azonosítja, ha annak pontjai és vonala:

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



Mosolygós arc

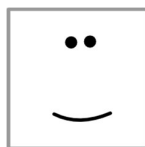


Nem mosolygó arc
A száj metszi a piros vonalat.

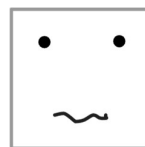
Az alkalmazás 4 képről készített modellt. Ezekből egyet azonosított mosolygós arcként. Melyiket?



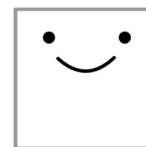
A)



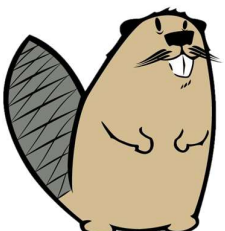
B)



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.

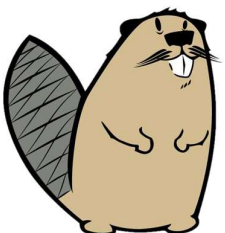
Forrás: <http://www.telegraph.co.uk/technology/2016/12/07/robot-passport-checker-rejects-asian-mans-photo-having-eyes/>

KULCSSZAVAK

Képfelismerés, algoritmus alkalmazása, gépi látás

WEBOLDALAK

https://hu.wikipedia.org/wiki/Gépi_látás



SZOMSZÉDOK (2013-AT-11)

KADÉT – NEHÉZ

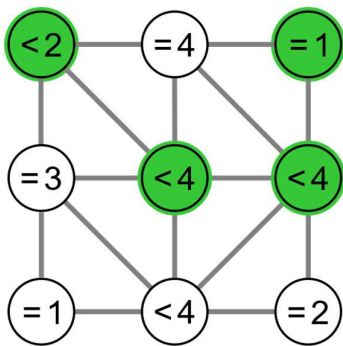
Az alábbi képen kilenc kör látható, amelyek részben össze vannak kötve.

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.

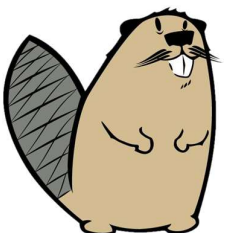
Már előre megjelöltünk néhány kiválasztott kört.

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



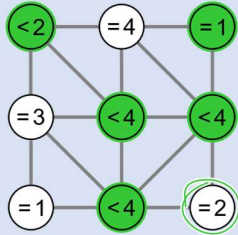
Melyiket?

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

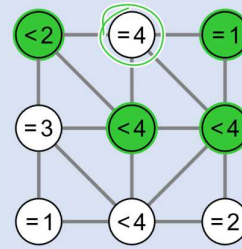


„A” válasz a helyes

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

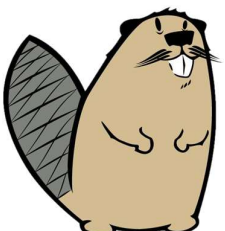
Érdekes azonban ebben az esetben logikusan és konzekvensen eljárni. Először azokat a köröket keressük meg, amikre a feltétel egyértelműen teljesülhet. Például minden kör, amelynél a feltétel „=n” és pontosan n kapcsolata, szóval n szomszédja van. Innen tudunk majd logikus következtetéssel továbbmenni: 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))



SZÁMÍTÓGÉPES JÁTÉK (2018-CH-03)

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.

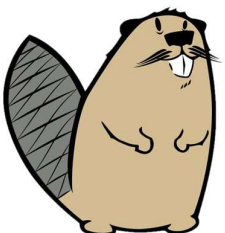
A játék hossza függ a játékkörök számától.

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

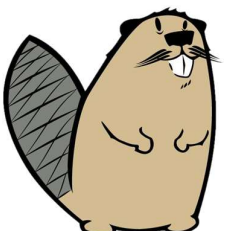
KULCSSZAVAK

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)

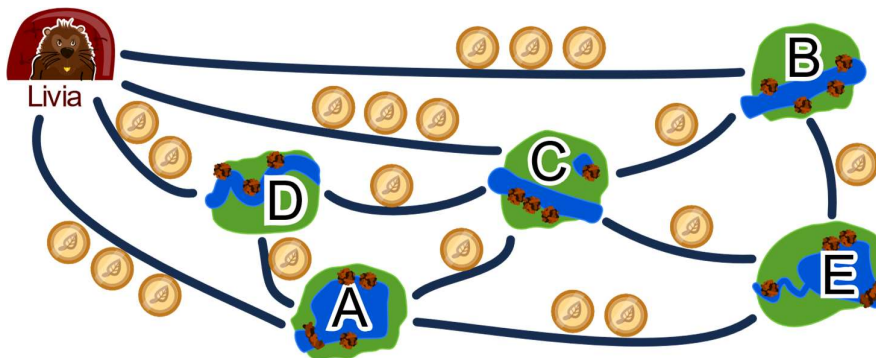
KADÉT – NEHÉZ

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

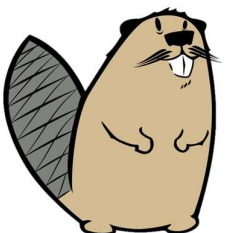
Mindenkit egy utazással látogat meg úgy, hogy nem megy el egynél többször ugyanabba a faluba és az utazás végén hazaér. Az egyes útszakaszok árát a képen láthatod.

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.

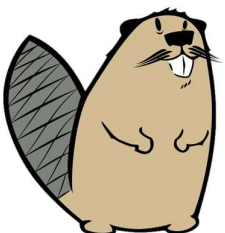
Ha egy-egy csomópontba nem csak egyszer mehetünk, a feladat tovább bonyolódik, hiszen még több lehetőségünk lesz a megoldásra.

KULCSSZAVAK

Optimalizáció, Utazóügynök probléma

WEBOLDALAK

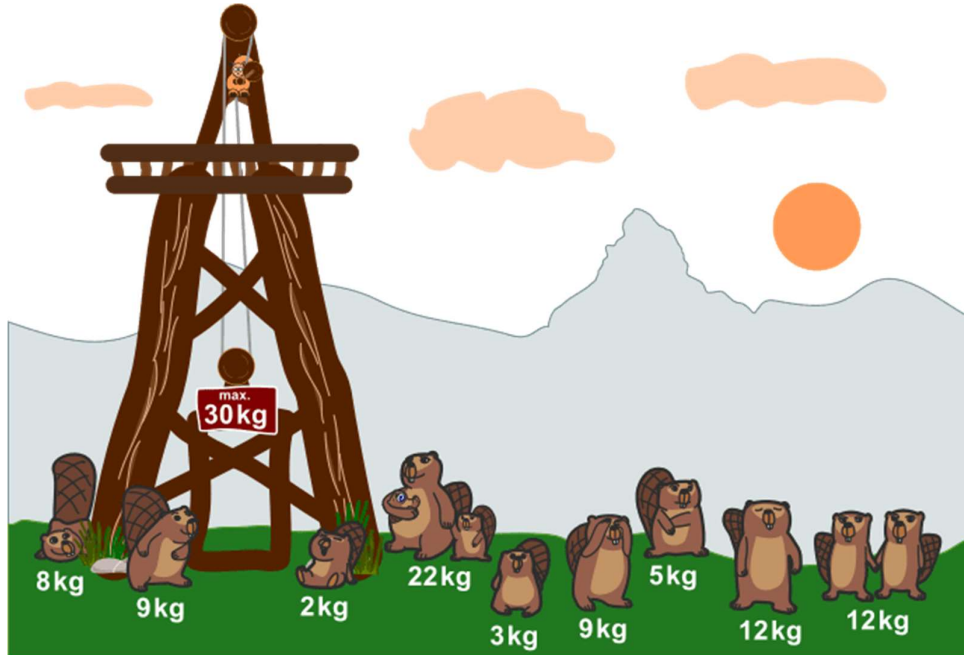
https://hu.wikipedia.org/wiki/Az_utazó_ügynök_problémája



FELVONÓ A MATTERHORN-ON (2018-CH-07)

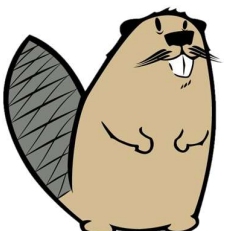
KADÉT – KÖZEPES

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 fordulóban, hogy a lehető legtöbb hódcsalád élvezhesse a kilátást?

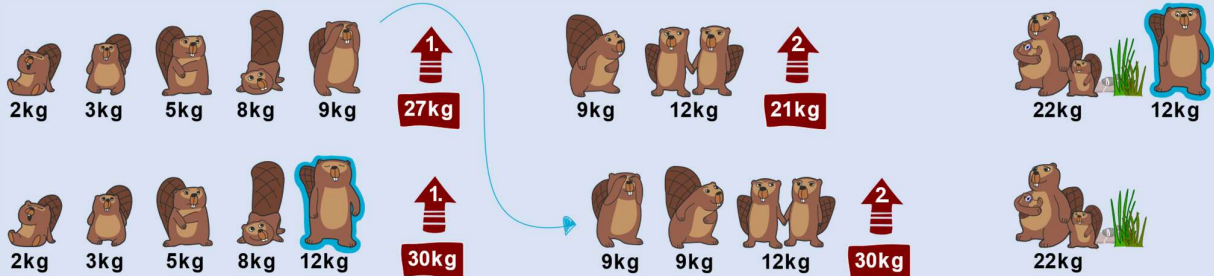
- A. Mindkét fordulóban 30-30kg.
- B. Az egyik fordulóban 27kg, a másikban 24kg.
- C. Az egyik fordulóban 30kg, a másikban 27kg.
- D. Az egyik fordulóban 27kg, a másikban 21kg.



„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.

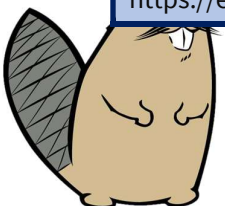
Ezt a problémát „Hátizsák-problémának” is nevezik, mivel meglehetősen sok objektumot kell bepakolnunk anélkül, hogy a maximális súlykorlátot elérnénk. Az ilyen és ehhez hasonló problémákat NP-teljesnek hívják. Csak megközelítőleg megoldhatóak, ami azt jelenti, hogy egy lehetséges és jó megoldást találhatunk,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

KULCSSZAVAK

Optimalizálás, hátizsák probléma

WEBOLDALAK

https://en.wikipedia.org/wiki/Knapsack_problem



Z BOLYGÓ (2018-DE-02)

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.  →  

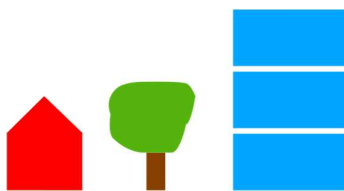
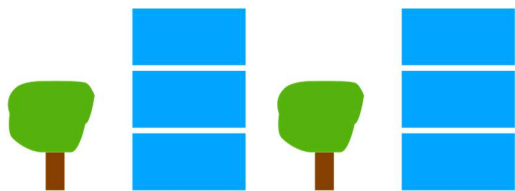
3.  →  

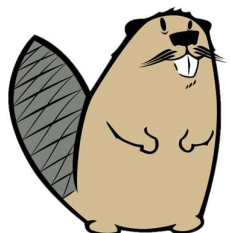
Például: ha először az 1., majd a 2. és végül kétszer a 3. szabályt alkalmazzák, egy olyan város alakul ki, mint a kép jobb oldalán látható:



Figyeld meg, hogy az egyes objektumok sorrendje nem változik.

Vajon melyik város NEM a Z bolygón található?

A. 	B. 
C. 	D. 



„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 építéskor 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.

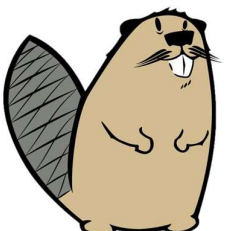
Minden programozási nyelvnek megvan a saját nyelvtana. Ez a nyelvtan határozza meg a programozási nyelv szintaxisát. Amikor a számítógép végrehajt egy programot, először azt vizsgálja meg, hogy a szöveg a programozási nyelven íródott-e (azaz a szintaxisát). Ehhez a programozási nyelv nyelvtanát használja.

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)

KADÉT – KÖZEPES

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

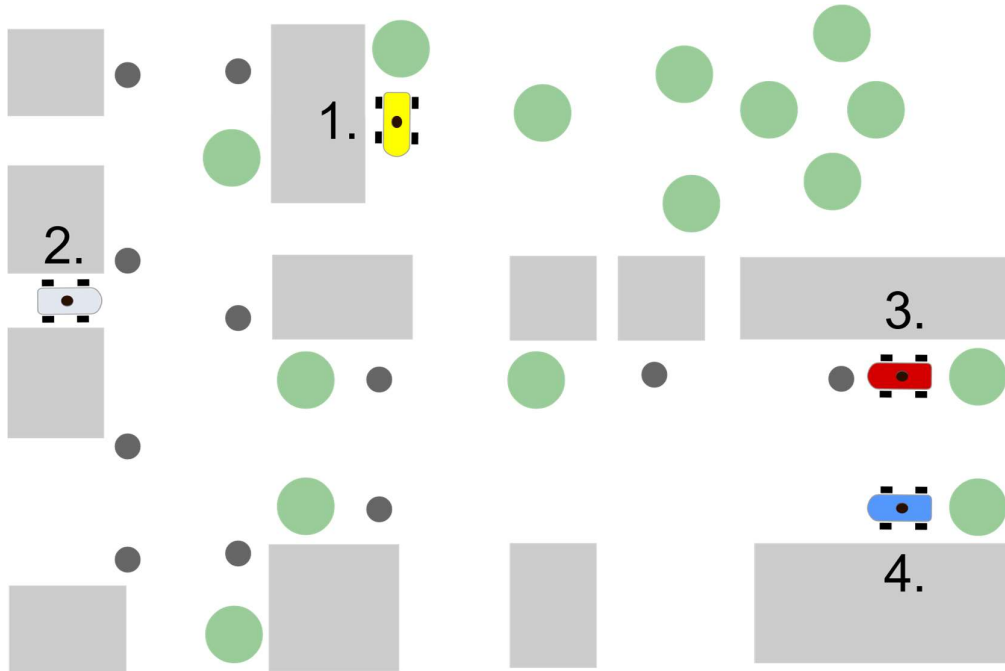
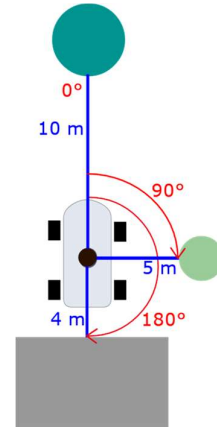
Röviddel azelőtt, hogy lemerült volna az akkumulátora, még sikerült egy parkolóhelyet találnia és modellt küldenie a környezetéről. Az elküldött modellben minden tárgyat, amit egy 360°-os érzékelővel elér, két értékkel ír le:

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

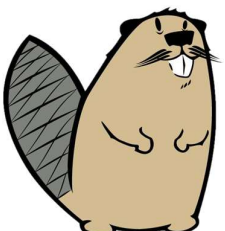
A példában például a modell: $[(0, 10), (90, 5), (180, 4)]$

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.

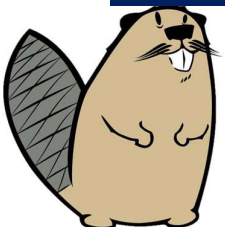
Általában egy modell a valóság egy leegyszerűsített leképezése, ahol csak azokat a szempontokat vesszük figyelembe, melyek valamilyen meghatározott célból fontosak számunkra. Az önvezető járművek esetében a LIDAR-modellt használják az összeütközések elkerülésére. Ebből a szempontból egyedül a tárgyak 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)**KADÉT – KÖNNYŰ**

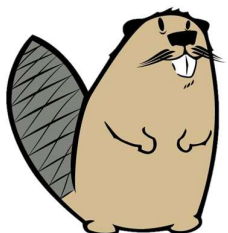
Karesz a következő három szabályt alkalmazza, amikor jelszót választ:

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. 😊

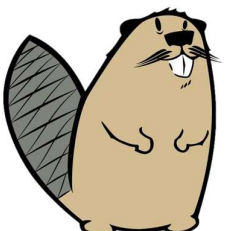
KULCSSZAVAK

Jelszó, biztonság

WEBOLDALAK

https://hu.wikipedia.org/wiki/Brute_force-támadás

https://hu.wikipedia.org/wiki/National_Institute_of_Standards_and_Technology



A HAZÚGOT HAMAR UTOLÉRIK (2018-HR-05)

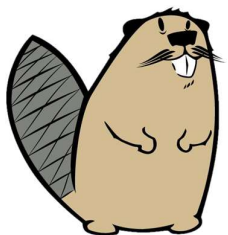
KADÉT – KÖZEPES

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 gyerekénél 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 Boole (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.

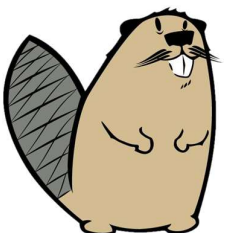
Vannak azonban számítógépek, melyek alapját más rendszer képezi. Például az 1950-es évek végén a három-értékű logikára (Lukasiewicz lengyel tudós ötlete) épült angol kezdeményezésű és orosz megvalósítású számítógép (ternary computer).

KULCSSZAVAK

Logika, kizárt harmadik

WEBOLDALAK

https://hu.wikipedia.org/wiki/Kizárt_harmadik_elve
https://hu.wikipedia.org/wiki/George_Boole
<https://hu.wikipedia.org/wiki/Boole-algebra>
<https://hu.wikipedia.org/wiki/Implikáció>
https://hu.wikipedia.org/wiki/Többértékű_logika
http://www.hiradastechnika.hu/data/upload/file/1978/01/1978_01_03.PDF

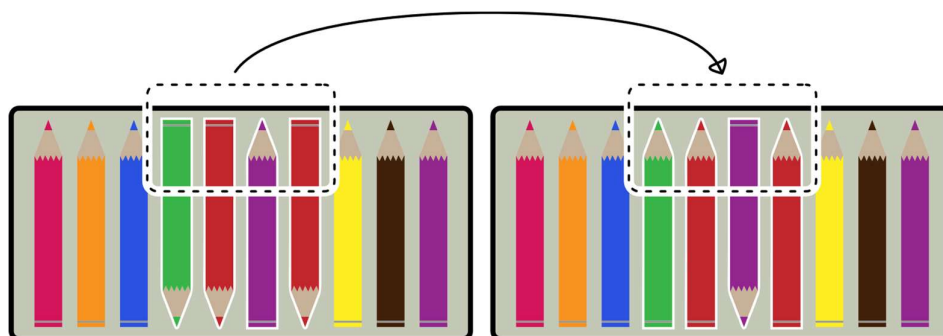


ADA CERUZÁI (2018-HU-01)

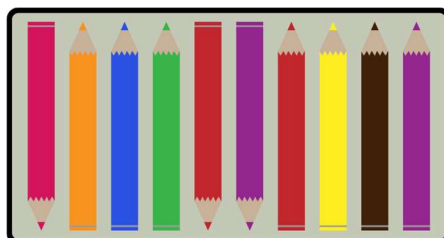
KADÉT – KÖNNYŰ

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

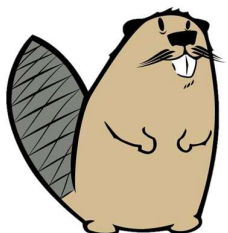
Egy fordítással egyszerre kettő vagy több, egymás mellett lévő ceruza irányát tudja megváltoztatni.



Most így állnak a dobozban a ceruzák:



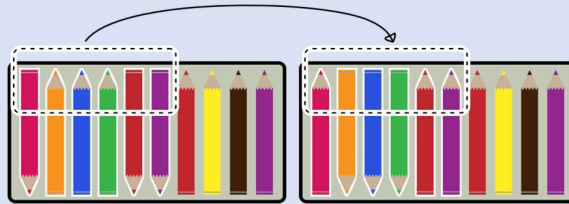
Legkevesebb hány fordítással tudja Ada megfordítani a ceruzáit, hogy mindegyik hegygel felfele nézzen?



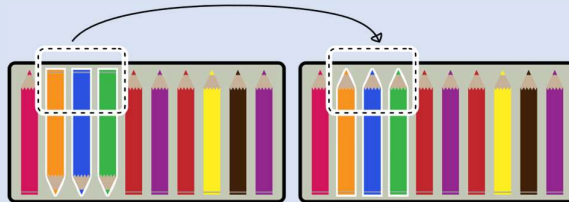
„2” a helyes válasz

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 negyedikig 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 tár)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>

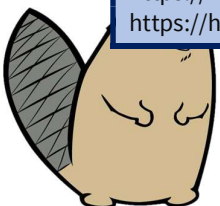
KULCSSZAVAK

Hatékonyság, memória

WEBOLDALAK

https://hu.wikipedia.org/wiki/Tartós_állapotú_meghajtó

[https://hu.wikipedia.org/wiki/Memória_\(számítástechnika\)#Blokk_vagy_fájl_hozzáférésű_memória](https://hu.wikipedia.org/wiki/Memória_(számítástechnika)#Blokk_vagy_fájl_hozzáférésű_memória)



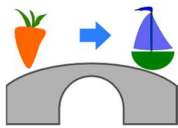
VÍZESÉS (2018-HU-03)

KADÉT – KÖZEPES

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.

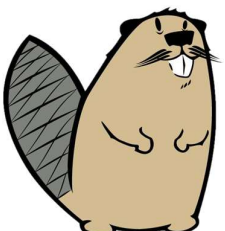
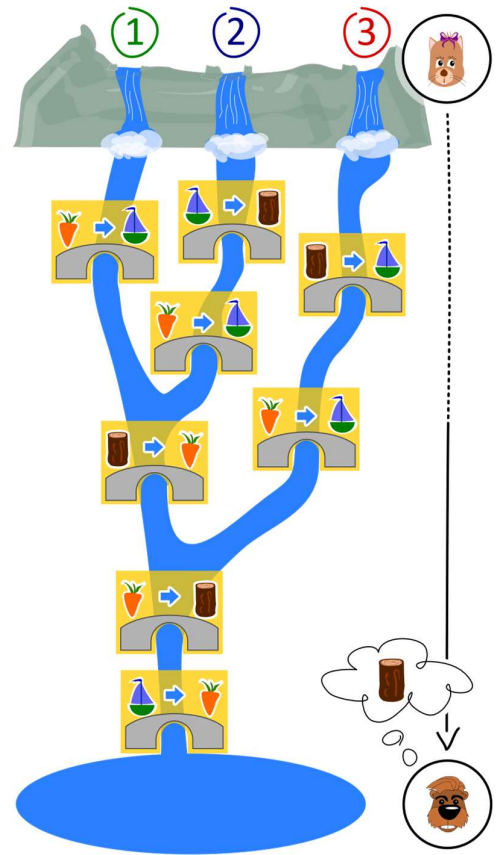


Például, ha egy répa egy ilyen híd alatt sodródik át, kicserélik azt egy kishajóra.

János a tónál áll.

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

- Egy kishajót  az 1. vízesésbe.
- Egy kishajót  a 2. vízesésbe.
- Egy répát  a 2. vízesésbe.
- 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.

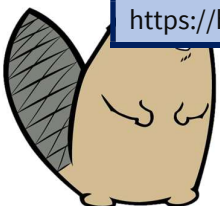
Ebben a hód-feladatban a hidak (a trollokkal) olyanok, mint egy-egy kis függvény, és az egész rendszer maga a program, amit funkcionális programnyelven írtunk meg.

KULCSSZAVAK

Funkcionális programozás, programozási paradigma

WEBOLDALAK

https://hu.wikipedia.org/wiki/Programozási_paradigma
https://hu.wikipedia.org/wiki/Funkcionális_programozás

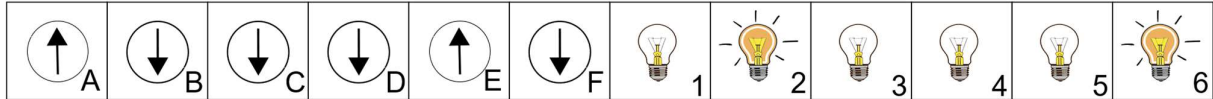
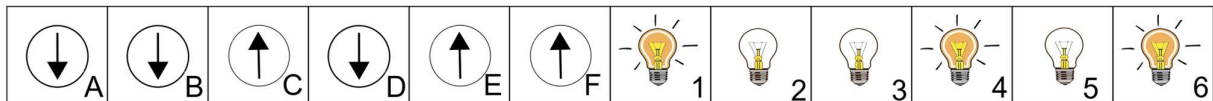
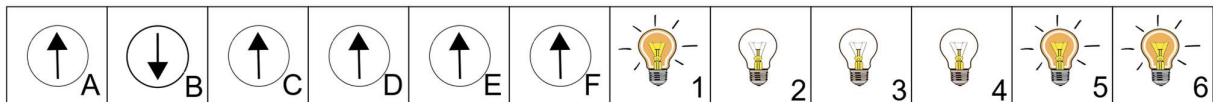
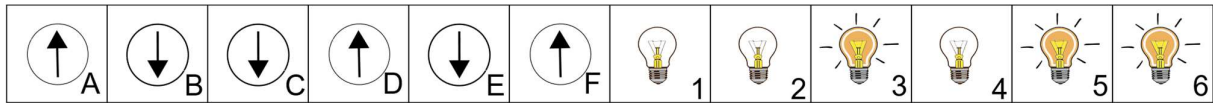


LÁMPÁK ÉS KAPCSOLÓK (2018-HU-08)

KADÉT – NEHÉZ

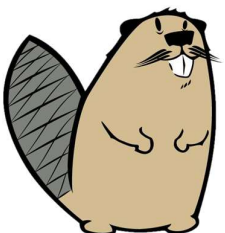
Tamás összekötött 6 lámpát 6 kapcsolóval. Minden kapcsoló pontosan egy lámpát kapcsol fel vagy le, de nem tudjuk, melyik kapcsoló melyik lámpához tartozik. A kapcsolók egy része akkor kapcsolja fel a hozzá kötött lámpát, amikor a nyíl felfele mutat, a többinél pedig fordítva.

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.

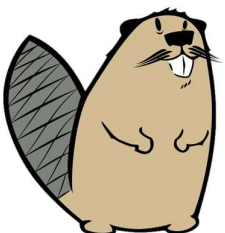
KULCSSZAVAK

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



HÓD KAPITÁNY NAPJA (2018-KR-08)

KADÉT – NEHÉZ

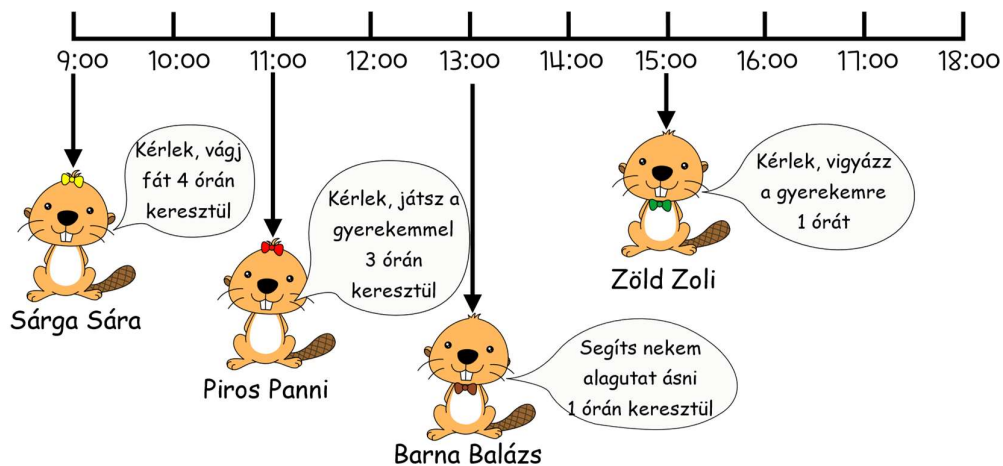
Hód kapitány szívesen segít, amikor egy másik hód megkéri valamire. Sajnos egyszerre csak egy feladatot tud megcsinálni. Amikor Hód kapitányt akkor kérik meg, hogy segítsen, amikor épp valami mást csinál, az alábbi fontossági táblázat alapján dönt:

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

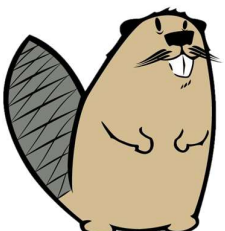
Amikor valaki segítséget kér Hód kapitánytól és az új feladat

- 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.

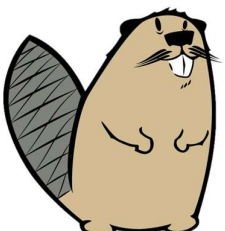
Ahogy feladat is mutatja, ilyen helyzet a mindennapi életben is előfordulhat, főleg akkor, amikor több szereplő kevés erőforrásért versenyez.

KULCSSZAVAK

Feladat ütemezés, szál (processz)

WEBOLDALAK

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



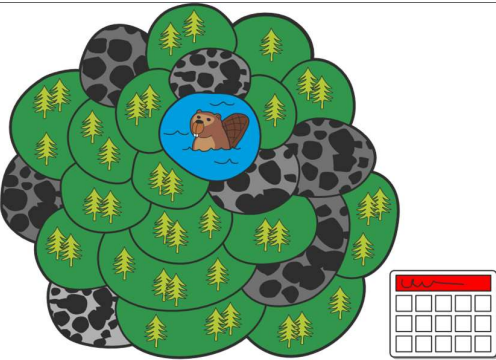
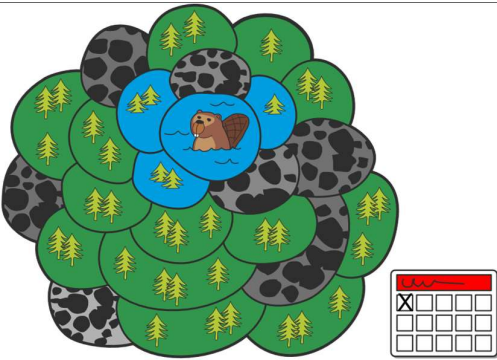
HÓD TÓ (2018-LT-02)

KADÉT – KÖZEPES

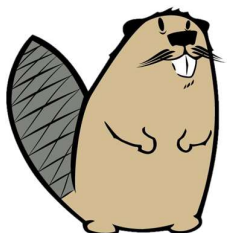
Egy völgyben, ahol hódok élnek van egy kis tó. Erdős és sziklás területek veszik körül.

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.

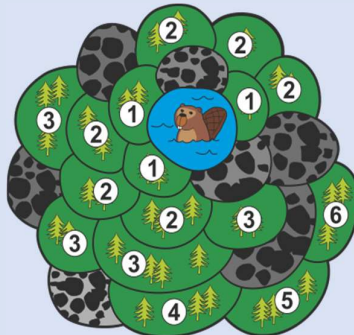
A völgy eredetileg	1. nap után
	

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. Fiatalok egy olyan csoportja, ahol mindenki legalább egy emberrel barátságban van a csoporton belül is egy összefüggő terület, ha a barátságokat mint szomszédságot tekintjük.

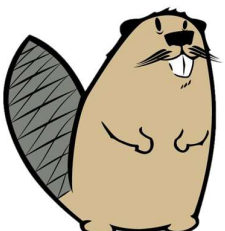
Az informatika ismer módszereket, amivel megállapíthatjuk, hogy egy terület összefüggő-e, mint például a szélességi vagy a mélységi keresés. Ezeknek a módszereknek a segítségével átszínezhettük egy kép egy részét vagy csoportot hozhatunk létre egy szociális hálón belül.

KULCSSZAVAK

Gráf, szomszédsági gráf, gráf bejárás

WEBOLDALAK

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



29-ES HÁZIKÓ (2018-SK-07)

KADÉT – NEHÉZ

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

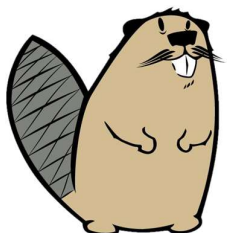
Egyes házakon már szerepelnek számok. A munkát az 50-es számnál kezdi és a 29-es tábla van a kezében. A következő utasításokat kapta a főnökétől:

- 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.

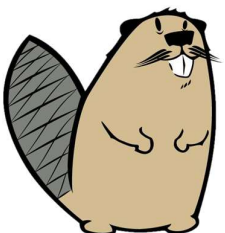
Mikor vissza akarunk keresni egy elemet, könnyen el tudjuk dönteni melyik utat válasszuk.

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)

KADÉT – KÖZEPES

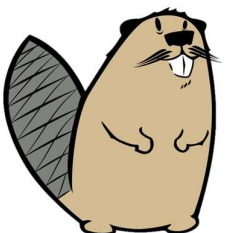
Kaptál egy üzenetet az Suli Menedzsment RendSzerből a következő tárggyal: „Frissítsd az adataidat!”.

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.

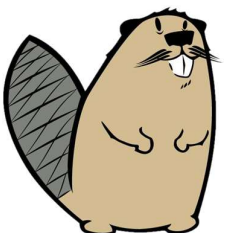
Ez ellen természetesen léteznek már szűrő-alkalmazások is, de fontos, hogy te is figyelj arra, hogyan kezeled a bizalmas adataidat (mint például a jelszavaid)

KULCSSZAVAK

Biztonság, adathalászat

WEBOLDALAK

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

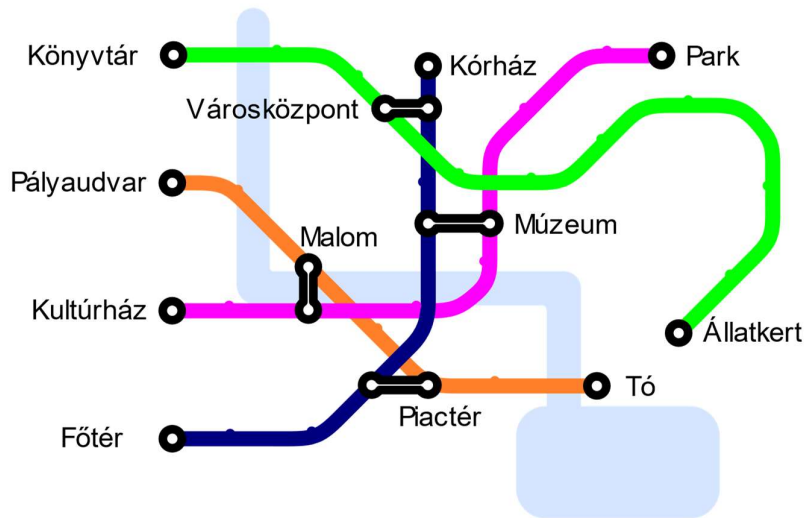


ÚTHÁLÓZAT (2013-TW-02)

KADÉT – KÖNNYŰ

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

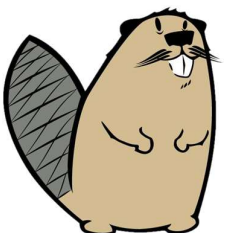
Négy olyan megálló van, ahol az egyik metróvonalról a másikra átszállhatunk: „Városvközpont”, „Múzeum”, „Malom” és „Piacér”.



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

Vajon melyik metróvonalon kezd meg az utazását?

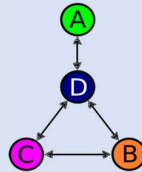
- A. 4 süteményt.
- B. 5 süteményt.
- C. 6 süteményt.
- D. 7 süteményt.



„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).

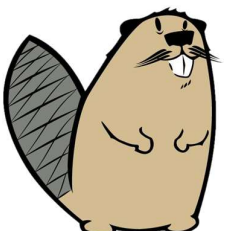
Gráfokat még több helyen használnak. Emberek szociális kapcsolatainak ábrázolására, útvonaltervezéshez vagy vásárlási ajánlatok/javaslatok elkészítésére. Tehát az is fontos informatikával kapcsolatos kompetencia, hogy tudj gráfokkal bánni 😊

KULCSSZAVAK

Gráf, közlekedési hálózat, térkép

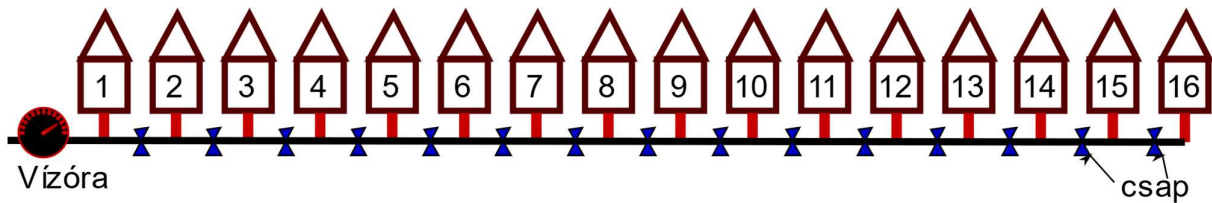
WEBOLDALAK

<https://hu.wikipedia.org/wiki/Gráf>



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.

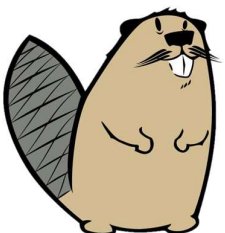
A vizsgálatot a következőképpen folytatjuk le:

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.

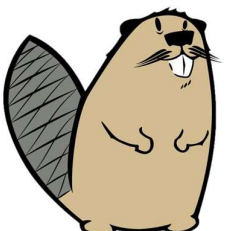
A bináris keresés esetében a vizsgálatok száma valóban jóval kevesebb, mint ha úgy keresnénk, hogy minden elemet sorban megvizsgálunk. 1000 könyv esetében a bináris kereséssel csak 10 keresési lépést szükséges végrehajtanunk.

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.

