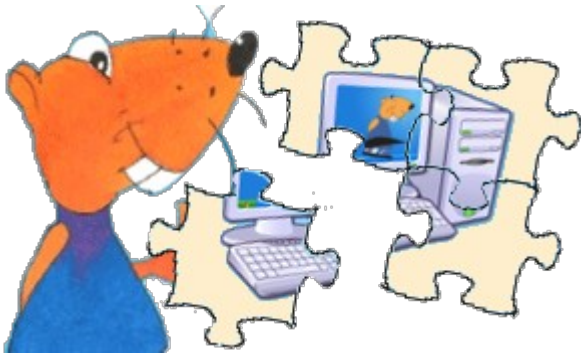




**Bebras – Beaver – Beber – Bever – Bobr – Bobor –
Bobřík – Kobras – Majava – Castoro – Castor –
Бобер – Hód**

**Bebras: Nemzetközi informatikai és számítógép-
készség verseny (International Contest on
Informatics and Computer Fluency) –
MINDENKINEK**

Az *e-HÓD/HÓD*ítsd meg a biteket a BEBRAS-kezdemenyezés magyar partnere.

A Bebras Dr. Valentina Dagiene litván professzor által életre keltett verseny, mely a nemzetközi Kenguruhoz hasonló célokkal rendelkezik, de nem a matematika, hanem az informatika területén. Bebras litvánul hódót jelent.

A verseny 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 területének 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.

A versenyt négy korcsoport számára rendezik:

- 5. és 6. osztály, Benjamin
- 7. és 8. osztály, Meteor
- 9. és 10. osztály, Junior
- 11. és 12. osztály, Senior.

Magyarországon 2013-ban harmadik alkalommal, mind a négy korcsoportban meghirdettük a megmérettetést.

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

Az alábbi dokumentumban a 2013-as magyar verseny feladatai és megoldásai találhatók.

További információért Látogasson el a <http://e-hod.elte.hu/> weboldalra, vagy írjon e-mail-t az info@e-hod.elte.hu címre.

Résztétel

A résztétel mindenki számára ingyenes.

A verseny november második hetében kerül lebonyolításra, osztályonként kiválasztható, hogy az adott héten melyik napon mikor (reggel 8:00-tól délután 4-ig). Ezzel biztosítható, hogy akár egy-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 átbeszélhetőek ugyancsak akár egy tanóra keretein belül.

Szabályok

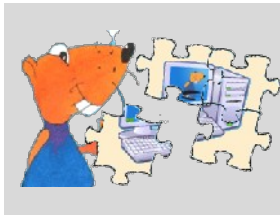
- a résztvevők online kapják meg és válaszolják meg a kérdéseket;
- a versenyre fordítandó idő 45 perc, 18 feladat három nehézségi szinten: könnyű, közepes és nehéz;
- a verseny alatt semmilyen más számítógépes program, alkalmazás nem használható;
- a verseny során nyugalmas környezetet kell biztosítani;
- 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 megválaszolhatóak;
- a kérdések, problémák megértése a feladat részét képezi. Ezért a feladatok megbeszélése, értelmezéssel kapcsolatos kérdések nem megengedettek;
- a verseny befejezése után, a hód hetet követően kerülnek nyilvánosságra a megoldások;

Értékelés, pontozás

Minden 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

Minden résztvevő kezdetben 54 pontot kap. Így összesen maximum 216 pontot érhet el, illetve 0-ra csökkentheti pontjait, amennyiben minden kérdésre helytelen választ adott.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Iskolai kirándulás (2008-DE-09)

Sajnos a legutóbbi informatika óráról hiányoztál.

Azonban akkor beszélte meg az osztály a következő kirándulást, ami egy számítógép-múzeum meglátogatása.

E-mailben kéred a tanárnódet, hogy küldje el neked a kirándulás programját.

Melyik lenne a megfelelő tárgya („subject”) az e-mailnek?

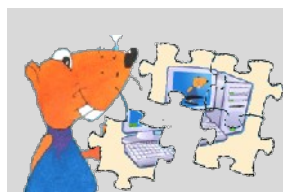
- A.) Hír rólam
- B.) Sürgős!
- C.) Iskolai kirándulás a számítógép-múzeumba
- D.) Meg szeretném kérni, hogy a kirándulás programját az infókkal küldje el nekem – köszönöm széééééépen...

„C” válasz a helyes:

Az „A” válasz a küldőről ad információt, nem a tárgyról. A „B” válasz semmiféle információt nem tartalmaz az e-mail tartalmáról. A „D” válasz egy rövid cím helyett maga az e-mail.

Ez informatika!

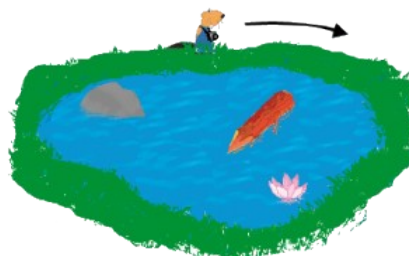
Az emberek közötti értelmes kommunikáció már akkor is rendezett és strukturált volt, amikor még nem voltak számítógépek és okostelefonok. A szoftveres kommunikációs eszközök, mint például az e-mail, igyekeznek ezeket a rendszereket és a struktúrákat követni. Amennyiben a felhasználó nem tartja magát az írott, illetve íratlan szabályokhoz (például a netikett), nem képes megfelelően kommunikálni.



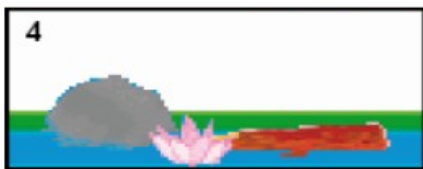
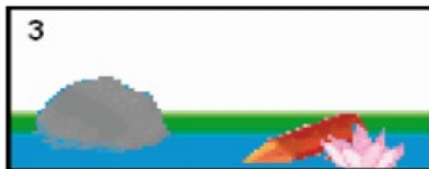
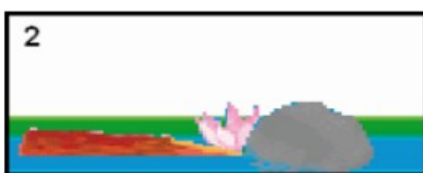
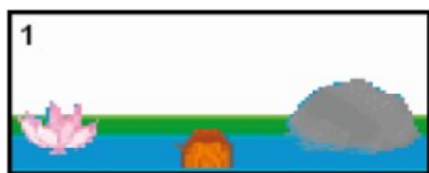
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Fotózás (2010-LT-05)

Hód Henrik körbesétálta a tavat, ahol lakik. A képen jelölt ponttól indult a nyíl irányába.



A sétája alatt négy fényképet készített:



Milyen sorrendben készítette a fényképeket?

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

„B” válasz a helyes:

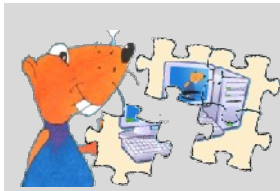
A mellékelt ábra mutatja, honnan készítette a fényképeket.



Ez informatika!

„Egy kép többet mond minden szónál”, és ha több összetartozó képünk van, akkor a vizsgálat még érdekesebb. Jelenleg a számítógépek, programjaik nem tudnak ez emberhez hasonló módon látni és a látottakat értelmezni, de az informatika több ága is foglalkozik ezzel a problémával és igyekszik az „intelligens” képértelmezést is megoldani.

Emellett természetesen az algoritmikus gondolkodás is szerepet kap a feladatban.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Színes gyöngynyálancok (2011-DE-14)

A kreatív hódasszony, Judit, gyermekei nyálancot fűznek. Különböző alakú gyöngyeik vannak (négyzet alakú és kör alakú), melyek kék vagy piros színűek.

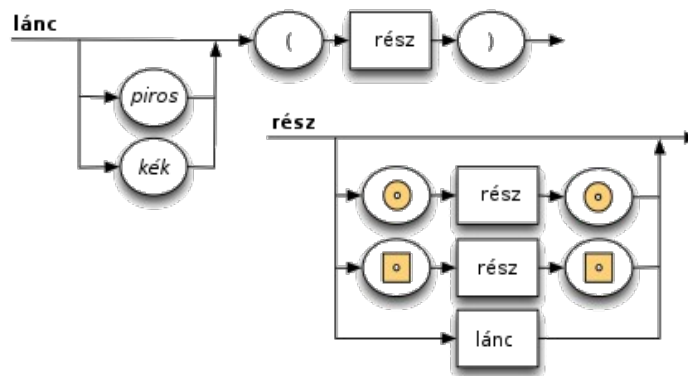
Így például a következő láncot fűzhetik:



Judit elmagyarázza a gyerekeinek, hogy ez a lánc a következőképpen írható le:

piros((*kék* ())

Judit két rajzot készít: ezeket "lánc"-nak és "rész"-nek hívja.



Csak olyan láncokat szeretne, melyek leírása a nyílatkat követve megvalósítható.

A kis hódok négy láncot is készítettek, de sajnos csak egy felel meg Judit rajzának. **Melyik?**



A.)



B.)



C.)



D.)

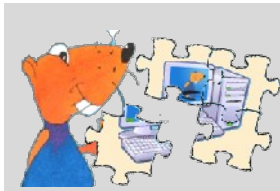
„D” válasz a helyes:

A jelölések Die Zeichnungen ergeben verschachtelte Beschreibungen von (möglicherweise gefärbten) Kettenteilen. Egy láncrész mindig ugyanolyan típusú gyönggyel kezdődik és végződik, melyek ugyanolyan színűek is. Minden lánc, amelyik a leírásnak megfelel, felosztható két részre, melyek egymásnak tükörképei. Ez csak a D válaszra igaz.

Az „A” válaszban a színek nem tükörképei egymásnak. A „B” válasznál a lánc közepén a kék kör és négyzet nem tükrösek. A „C” válasz esetében a középen található kék körnek nincs párja, így tükörképe sincs.

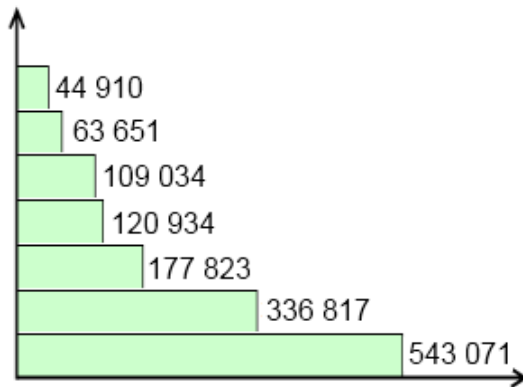
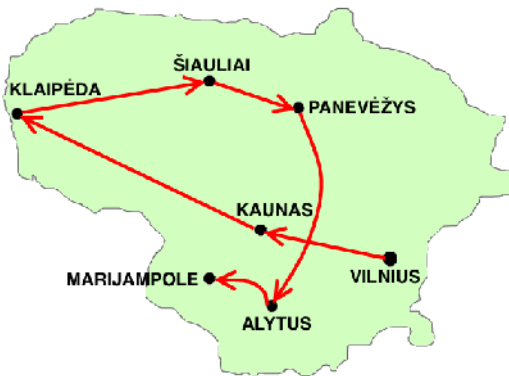
Ez informatika!

Judit hód jelöléseit az informatikában Szintaxis diagrammnak nevezzük. Egy programozási nyelv nyelvi szabályai ilyen szintaxis diagrammokkal kerül leírásra, meghatározásra.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Városok (2011-LT-14)



A térképen (balra) a legjelentősebb litvániai városokat átszelő utazást ábrázoltunk.

Az utazás a legnépesebb (543 071 lakosú) várossal, Vilniusszal kezdődik, majd onnan csökkenő sorrendben vezet a legalacsonyabb népességű városig.

Az oszlopdiagram (jobbra) a városok népességét mutatja. A városok nevei azonban hiányoznak.

Hány lakosú Alytus?

- A.) 44 910
- B.) 109 034
- C.) 336 817
- D.) 63 651

„D” válasz a helyes:

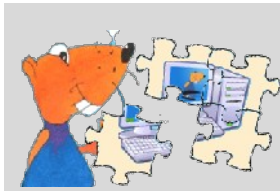
Alytus az utazás utolsó előtti városa. Népességszáma tehát a diagram felülről második oszlopában található.

Alytus a második legalacsonyabb népességű város.

Ez informatika!

Ha az adatok ügyesen vannak összekapcsolva, gyakran érdekes információkat nyerhetünk belőlük.

Itt az adatokat az útvonal, a lakosság száma és az útvonal sajátos tulajdonságai adják. Az emberek jól tájékozódnak, ha az adatmennyiség nem túl nagy és a diagram átlátható formában van. De a számítógépek az információk jóval nagyobb adatmennyiségével is elboldogulnak. Ehhez azonban az adatokat olyan megfelelő struktúrákban kell tárolnunk, mint pl. a „relációs adatbázisok”.



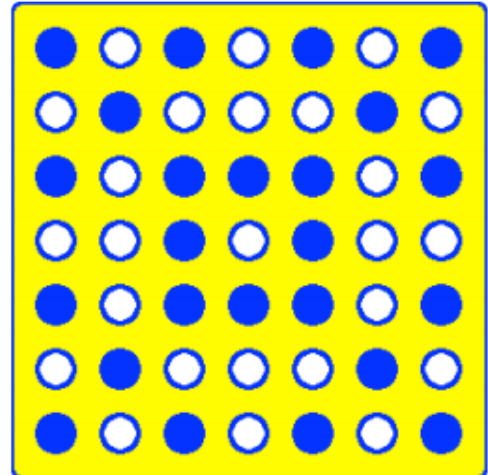
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Hotelkulcsok (2011-NL-01)

A Hód Hotelben új zárrendszert vezetnek be.

A vendég egy négyzet alakú 7-szer 7 kódponthoz tartozó műanyag kártyát kap. Minden kódpont lehet lyukas vagy nem lyukas. A képen egy ilyen kártya látható.

A szoba zárja egy kódolvasó. A műanyag kártya kódolása előről, hátulról, hosszában és keresztben is szimmetrikus. Így nem számít, hogy a vendég milyen irányba helyezi be a kártyát a zárba.

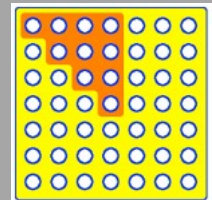


Hányféle kulcs készülhet így?

- A.) 16
- B.) 49
- C.) 1024
- D.) 65536

„C” válasz a helyes:

A négy lehetséges szimmetrikus állapot miatt csak egy 10 kódpontból álló rész különbözik. Az összes többi kódpont visszavezethető ezekre. Minden kódpont két állapotú: lyuk vagy nem lyuk, ez 2 a tizediken, azaz 1024 lehetséges kódot eredményez.



Ez informatika!

Az informatikai rendszereket (általában) emberek használják, akik időnként hibát követnek el. Ezért egy rendszer akkor használható igazán, ha többek között hibátűrő is. Az ebben a példában vázolt informatikai rendszer hibátűrő, hiszen mindegy, hogy a kulcs milyen irányban kerül a zárba. Ezt az úgynevezett redundanciával értük el: a zárkódokat egyszerre nyolcszor tartalmazza a kulcs.

	benjamin	nehéz	közepes	könnyű
	kadét	nehéz	közepes	könnyű
	junior	nehéz	közepes	könnyű
	senior	nehéz	közepes	könnyű

Moziban (2012-DE-02)

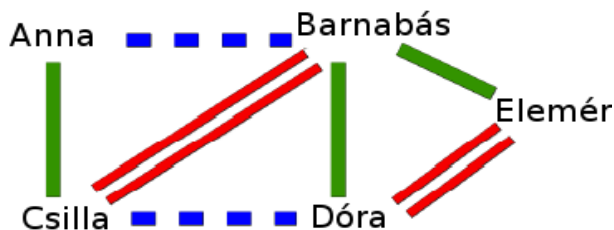
Anna, Barnabás, Csilla, Dóra és Elemér szeretnek moziba járni. Ott mindig együtt, egy sorban akarnak ülni.

Egy sorban azonban legfeljebb két másik ember mellett lehet ülni.

Ha ketten között valamilyen kapcsolat van, és egymás mellett ülnek a moziban, akkor érzik magukat a legjobban. A párok jó közérzetéből adódik össze a csoport jó közérzete.

<i>kapcsolat-jelölő</i>	<i>kapcsolat megnevezése</i>	<i>közérzetpont</i>
	ismerős	+1
	barát	+2
	szerelmes	+3

Az alábbi kép mutatja a kapcsolataikat:



Milyen sorrendbe üljenek le a moziban, hogy a csoport jó közérzete a lehető legmagasabb legyen?

- A.) Anna - Csilla - Barnabás - Elemér - Dóra
- B.) Anna - Barnabás - Csilla - Elemér - Dóra
- C.) Dóra - Barnabás - Csilla - Elemér - Anna
- D.) Csilla - Barnabás - Elemér - Dóra - Anna

„A” válasz a helyes:

A következő 4 sorrend: [Anna (2) Csilla (3) Barnabás (2) Elemér (3) Dóra]; [Anna (2) Csilla (3) Barnabás (2) Dóra (3) Elemér]; [Dóra (3) Elemér (2) Barnabás (3) Csilla (2) Anna]; [Elemér (3) Dóra (2) Barnabás (3) Csilla (2) Anna] adja a lehető legmagasabb közérzet pontot, +10-et. Ebből csak az első szerepel a lehetséges megoldások között.

Még magasabb pont eléréséhez minimum 3 szerelmespárra lenne szükség, de a mi esetünkben csak 2 kapcsolatban van "szerelem".

A B esetben: $1+3+0+3=7$; C esetben: $2+3+0+0=5$; D esetben: $3+2+3+0=8$ lenne az eredmény.

Ez informatika!

Az informatikában sok módszernek, melyek egy probléma lényegi részét leírják gráf az eredménye. Egy helyzetből a „legjobbat” kihozni, egy úgynevezett optimalizációs probléma az adott gráfon. A megoldás időnként egy jó ábrából felismerhető, de alkalmanként szükséges lehet „nyers erőre” (brute force módszerek), vagy heurisztikákra; illetve ezek kombinációira.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Cserekereskedelem (2012-FR-06)

Egy hatalmas árvízben Hód Henrik mindenét elvesztette, csak egy fogkefeje maradt meg. Ezt a HÓDNeten el tudja cserélni valami másra, amit szintén tovább tud cserélni valami újabb dologra.

A célja, hogy addig csereberéljen, amíg újra nem lesz egy háza.

A HÓDNeten a következő ajánlatokat találta. Például Anna egy fogkeféért egy lufit adna cserébe.

Név	Keres	Ad érte
Anna	fogkefe	lufi
Béla	fogkefe	kosár
Cecília	lufi	hajó
Dániel	hajó	motor
Emil	lufi	bicikli
Franciska	kosár	hajó
Gusztáv	kosár	kutya
Helén	kutya	lufi
Imre	bicikli	lufi
János	kutya	szőnyeg
Kata	szőnyeg	motor
Lili	festmény	szőnyeg
Mónika	bicikli	motor
Norbert	szőnyeg	ház

Kikkel cseréljen Hód Henrik, hogy végül ismét legyen háza?

- A.) Anna - Cecília - Franciska - Gusztáv - Norbert
- B.) Anna - Cecília - Franciska - Gusztáv - János - Norbert
- C.) Béla - Gusztáv - János - Norbert
- D.) Béla - Franciska - Cecília - Helén - János - Norbert

„C” válasz a helyes:

Az A, B és D megoldásoknál Franciska nem cserélne Cecíliával, mivel mindketten hajót szeretnének kapni!

Ez informatika!

Az egész cserekereskedelmet leírhatjuk egy irányított gráffal. A gráf csomópontjai a cserében résztvevő tárgyak, a nyilak pedig a csereajánlatokat jelölik.

A gráf egy útja, követve a nyilakat az egyik csomópontból egy másikba, mutatja, hogyan tudunk több cserét egymás után végrehajtani/megkötni. Nem minden csomópont érhető el bármely más csomópontból egy úton, tehát nem cserélhető el minden tárgy egy tetszőleges másik tárgyra.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Hátlapok (2012-NL-02)

Aristo 4 kártyalapot tesz eléd.

Mindegyik lap egyik oldalán egy betű, a másik oldalán egy szám van.

Aristo azt állítja, ha a kártya egyik oldalán magánhangzó van, akkor a másikon páros szám. Mint tudjuk E magánhangzó, V mássalhangzó, 2 páros, 7 páratlan szám.



De tudod, hogy Aristo vajon igazat mondott-e?

Legkevesebb hány kártyát kell megfordítanod ahhoz, hogy bizonyíthasd, igazat mondott-e.

- A.) 1
- B.) 2
- C.) 3
- D.) 4

„B” válasz a helyes:

Az E-kártyát meg kell fordítanunk, hogy megnézzük, páros szám van-e a másik oldalán. Ha páratlan, akkor tudjuk, hogy Aristo nem mondott igazat.

A V-t nem kell megfordítanunk, hiszen Aristo semmit sem állított a mássalhangzóról, ezért nem érdekel minket az igazságtartalma.

A 2-est nem kell megfordítanunk, hiszen ha magánhangzó van a másik oldalán, akkor igaz az állítás. Ha mássalhangzó van a másik oldalán, akkor pedig nem érdekel minket (ld. V esete).

A 7-est meg kell fordítanunk, hiszen ha magánhangzó van a másik oldalán, akkor Aristo nem mondott igazat.

Ez informatika!

Nem nehéz egy számítógépet következtetésekre programozni: majdnem minden programozási nyelvben léteznek a ha... akkor... (if..then..) elágazások. Sőt néhány programozási nyelvben egy jóval elterjedtebb emberi logikai gondolkodási hiba is programozható: (IF (IF a THEN b) THEN (IF b THEN a)) nem logikus és nem igaz!

A feladat nem csak informatika, hanem pszichológia :) is:

Wason szelekciós tesztje: http://hu.wikipedia.org/wiki/Wason_szelekci%C3%B3s_feladat

programozás, logika, implikáció.



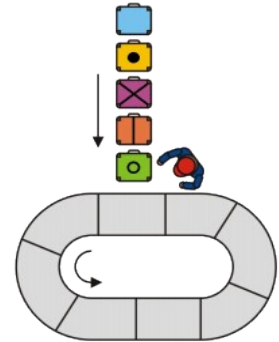
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Repülőtér (2013-AT-04)

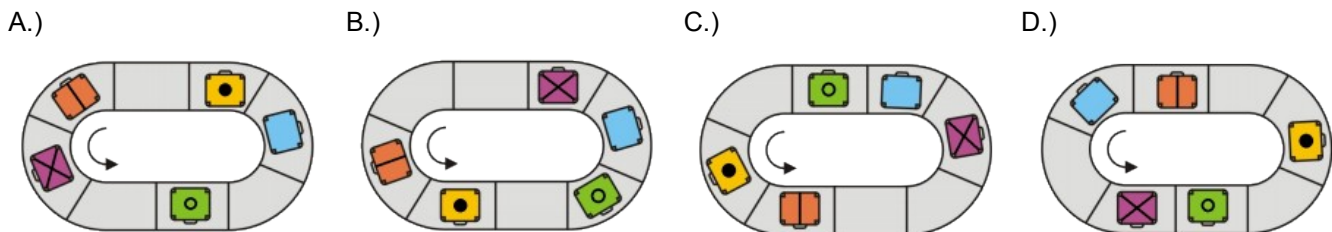
A repülőtéri szállítószalag a nyíl irányába forog körbe és 8 hely van rajta. Egy dolgozó 5 bőröndöt rak sorban a szállítószalagra.

A következő bőröndöt mindig a harmadik szabad helyre rakja, tehát a már foglalt helyeken kívül két szabad helyet hagy továbbfordulni.

Akkor van kész, ha mind az 5 bőröndöt felrakta a szállítószalagra.



Hogy néz ki a szállítószalag a munka végeztével?



„B” válasz a helyes:

Elsőként a körrel jelölt bőrönd kerül fel a szállítószalagra. Utána 3 hellyel hátrébb a függőleges vonallal, újabb 3 hellyel hátrébb pedig a kereszttel jelölt bőrönd.

Ezután a ponttal jelölt bőröndnek a harmadik szabad helyre kell kerülnie. Mivel azonban a körrel jelölt bőrönd ekkorra már körbeért, a harmadik szabad hely a kereszttel jelölt bőröndtől számított negyedik hely.

Utoljára a jelöletlen bőrönd kerül fel a szállítószalagra. Ennek a függőleges vonallal jelölt bőröndöt, két szabad helyet és a kereszttel jelölt bőröndöt kell elengednie.

Ez informatika!

Ha a táskák végrehajtandó műveleteket, a szállítószalagon lévő helyek pedig szabad kapacitást jelölnének, a feladat arról szólna, hogy hogyan lehet a végrehajtandó műveleteket a számítógép szabad teljesítményének függvényében elosztani (angolul „scheduling”).

A műveletek elosztását bonyolítja, hogy a feldolgozási idejük és a tárolásukhoz szükséges hely változó. Ráadásul a műveletek nem függetlenek egymástól, és különböző részrendszerekre, ill. csatlakozó eszközökre van szükségük, amelyek gyakran csak korlátozott számban állnak rendelkezésre. Érdemes megjegyezni, hogy az itt ábrázolt elosztás – minden harmadik szabad helyre kerül egy elem – nem a leghatékonyabb.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

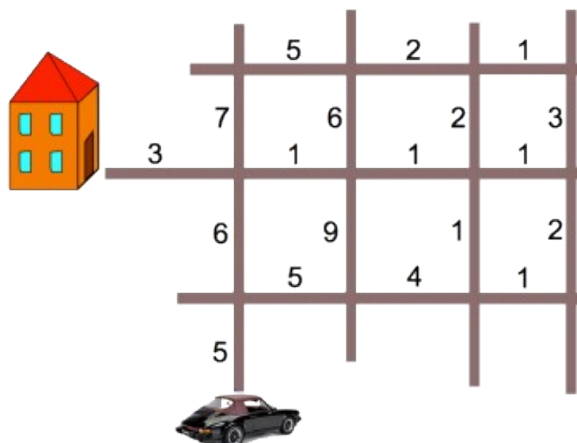
Sose fordulj balra! (2013-AT-08)

Végtelen szembeforgalom – gyakorlatilag lehetetlen ezeknél a kereszteződésnél balra fordulni. Ha az autós gyorsan haza akar jutni, egy olyan útszakaszt kell választania, ahol sosem kell balra fordulnia.

A képen percekben megadva láthatod azt az időtartamot, amire az autónak egy-egy útszakasz megtételéhez szüksége van.

Legkevesebb mennyi időre van szüksége az autónak, ahhoz hogy hazaérjen, ha sosem fordulhat balra?

- A.) 35 perc
- B.) 33 perc
- C.) 32 perc
- D.) 30 perc



„D” válasz a helyes:

A leggyorsabb út a D blokk, majd a C blokk megkerülésével történik.

Ehhez 30 percre van szükség:

$5+6+1+1+1+2+1$ a piros szakaszon

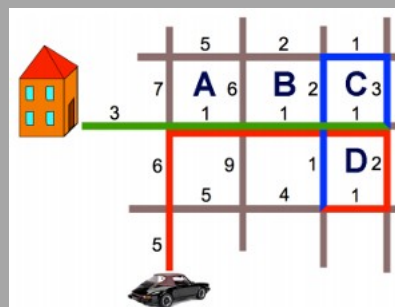
$1+2+1+3$ a kék szakaszon

és $1+1+1+3$ a zöld szakaszon.

Minden más úton hosszabb a menetidő. Az A blokk megkerüléséhez 33

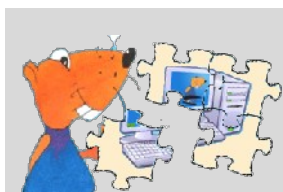
percre van szükség (A válasz). Az A és a B blokk megkerülése 32 percig tart (B válasz).

Az A, a B és a C blokk megkerülésére 35 percre van szükség (C válasz).



Ez informatika!

Az informatikában gyakori probléma a minimális ráfordítást igénylő út keresése, ráadásul sokszor bizonyos keretfeltételeket is figyelembe kell venni. A feladatban például ilyen keretfeltétel az egyes útszakaszok menetideje, illetve az a kikötés, hogy nem tudunk balra fordulni. Gyakran túl nagy a szóba jöhető utak száma ahhoz, hogy mindegyiket egyesével megvizsgálják és megállapítsák, hogy az-e a minimális ráfordítást igénylő út. Ilyenkor meg kell próbálni a megvizsgálandó utak számát ésszerűen szűkíteni. A feladatot néhány környékbeli háztömbre korlátoztuk. Ezzel kockára tesszük, hogy egy 30 percnél kevesebb időráfordítást igénylő utat találjunk, amely a mi kutatási horizontunkon (a képünkön) kívül vezet



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

A-ból C-be (2013-AT-09)

Van egy kis robotod, aki a következő parancsokat tudja végrehajtani:

E	Haladj egy lépést előre!
B(szög)	Fordulj balra! A szög nagysága zárójelben van megadva
J(szög)	Fordulj jobbra! A szög nagysága zárójelben van megadva

Kezdetben a kis robot mindig az A pontban áll és jobbra fordulva vár a programjára.

Ha a kis robotnak több parancsot kell egymásután végre hajtania, a parancsokat + jellel kapcsolják egymáshoz.

Például ez a program $E+B(20)+E+J(2)$ a következőt jelenti: a robot egy lépést előre lép, majd 20° -kal balra fordul, majd egy lépést előre lép és végül 2° -kal jobbra fordul.

Ha valamit a kis robotnak többször végre kell hajtania, azt *-gal jelöljük.

Például ez a program $180*(E+B(1))$ azt jelenti, hogy a robotnak egymás után 180-szor kell az E és a B(1) parancsot elvégeznie.

A kép a robot A pontból B pontba vezető útját ábrázolja.



Melyik programmal járja végig a kis robot az A-ból C-be vezető utat?



- A.) $90*(E+B(1)+E+J(1))$
- B.) $90*(E+B(1))+90*(E+J(1))$
- C.) $90*(E+B(1))+J(30)+90*(E+J(1))$
- D.) $B(90)+90*(E+B(1))+J(90)+90*(E+J(1))$

„B” válasz a helyes:

$90*(E+B(1))$ paranccsal a kis robot egy negyed kört tesz meg balra, majd a $90*(E+J(1))$ paranccsal egy negyed kört jobbra.

Az A programmal a robot egy kb. egyenesen úton jut el A-ból C-be.

A C programnál a J(30) programrésszel letérne az útról, és nem érné el a kis robot a C pontot.

A D program első parancsa B(90) folytán a kis robot nem balra indul, hanem fölfelé.

Ez informatika!

A robot egy kezdeti állapotból (A pozícióban, B pont felé fordulva) indul és programparancsokat hajt végre, amíg a kódok végére nem ér. Ha a megfelelő kódsorozatot adjuk a robotnak, eljuttathatjuk a kezdőpontból a célpontba. Ha egyetlen rossz parancs kerül a kódba, az a robotot rossz irányba vezeti.

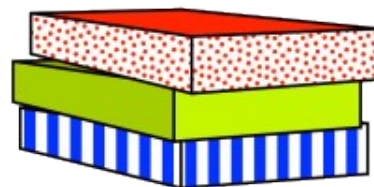


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

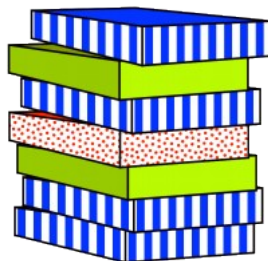
Süteményt elvitelre! (2013-AT-11)

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 hármaskupacot azonnal Tomihoz, az eladóhoz viszi. Tomi a hármaskupacot az eladásra szánt dobozok tetejére teszi. Ha Tomi elad egy süteményt, mindig a legfelső csomagot veszi le az eladásra szánt dobozok közül. Tibi gyorsabban süt, mint amilyen gyorsan Tomi eladja a süteményeket.



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 hármaskupacból sem hiányzik egy sütemény sem. Az is lehet, hogy Tomi nyolc vagy tizenegy süteményt is eladott már, hiszen ha egy hármaskupacból mindhárom süteményt eladta, azt nem tudjuk leolvasni az eladásra szánt csomagok kupacából.

Ez 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...

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



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Mágikus masina (2013-BE-07)

A mágikus masina golyókból és gombokból áll, a golyók érméket tartalmazhatnak. A golyók és a gombok nyilakkal vannak összekötve.

Az a golyó, melyből nyíl vezet egy gomb felé, az az adott gombnak a „forrása”. Az a golyó, melyhez egy gombból kiinduló nyíl vezet, az az adott gombnak a „célja”.

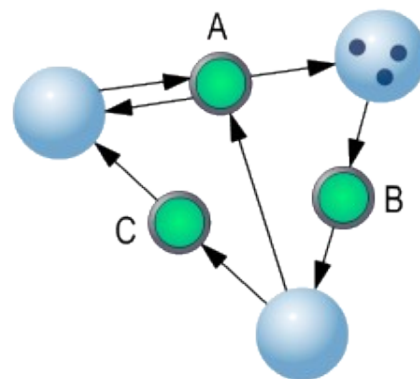
Ha megnyomsz egy gombot, egymás után két dolog történik:

(1) A masina megvizsgálja, hogy a gomb összes forrása tartalmaz-e legalább egy érmét.

(2) Ha tartalmaz, a gomb az összes forrásából eltüntet egy érmét és az összes céljához hozzáad egy-egy érmét.

Például, ha megnyomom a B gombot, a jobb felső golyóból eltűnik egy érme, a lentí golyóban pedig megjelenik egy érme.

Ha az adott gombokat a megfelelő sorrendben nyomjuk meg, a masina stabil állapotba kerül. Akármelyik gombot megnyomhatjuk, ez az állapot többé nem változik.



A gombnyomások melyik sorrendjével érhetjük el a stabil állapotot?

- A.) B – B – C – A – B – A
- B.) B – C – B – C – B – A
- C.) B – B – C – B – C – C
- D.) B – C – B – B – A – A

„C” válasz a helyes:

Az összes érmét a bal golyóba kell juttatnunk. Így nincs többé olyan gomb, melynek összes forrása legalább egy-egy érmét tartalmaz. Ebben a stabil állapotban egyik gomb megnyomása sem tud változást előidézni. Csak a C) válaszban jelölt sorrend hozza a masinát ebbe az állapotba.

Az A) válaszban jelölt sorrend a következő állapotot idézi elő: baloldalt 1, jobboldalt 2, alul semmi

Az B) válaszban jelölt sorrend a következő állapotot idézi elő: baloldalt 2, jobboldalt 1, alul semmi

Az D) válaszban jelölt sorrend a következő állapotot idézi elő: baloldalt 1, jobboldalt 2, alul semmi

Ezek közül az állapotok közül mindegyik esetben találunk érmét a jobb golyóban, így a B gomb megnyomása ezt az instabil állapotot meg tudja változtatni

Ez informatika!

A mi mágikus masinánk tulajdonképpen egy egyszerű Petri-háló leképezése. A Petri-háló egy párhuzamos rendszerek leírására használt formalizmus, melyet az 1960-as években Carl Adam Petri fejlesztett ki (<http://hu.wikipedia.org/wiki/Petri-h%C3%A1l%C3%B3>)

A Petri-hálót diszkrét dinamikus rendszerek modellezésére és szimulációjára használják, például bonyolult hivatalos eljárások és gyártási folyamatok során. Az informatikában a Petri-hálók hasznos eszközök a fentiekhez hasonló alkalmazási területeket szolgáló szoftverek kifejlesztésében és elemzésében.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

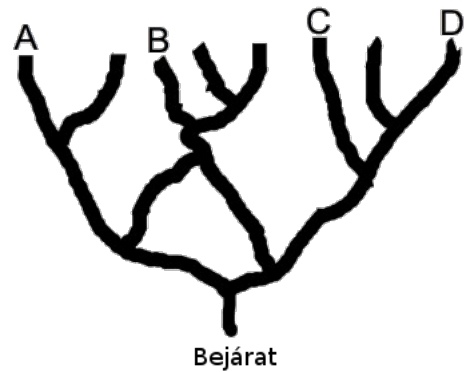
Barlangászok (2013-BE-09)

21 barlangkutató szeretne megvizsgálni egy barlangrendszert. A bejáratától indulnak, és az elágazásoknál egyre mélyebbre és mélyebbre haladnak a barlangrendszerben, egyre jobban eltávolodnak a bejáratától.

Egy-egy elágazásnál elosztják a csoportot: mindig ugyanannyi barlangász megy jobbra, mint balra. Ha páratlanul vannak, mindig egy személlyel több választja a jobb irányt.

Végül melyik helyszínre érkeznek a legtöbb barlangász?

- A.) Az A helyszínre.
- B.) A B helyszínre.
- C.) A C helyszínre.
- D.) A D helyszínre.

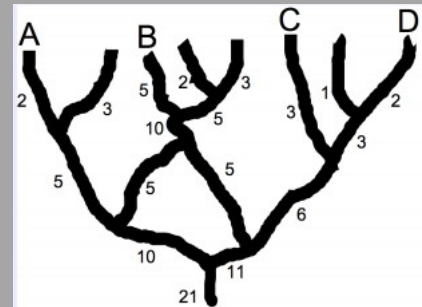


„B” válasz a helyes:

A képen láthatod a személyek megoszlását a különböző elágazásoknál.

Ez informatika!

Az elágazó barlangjáratokban folyó kutatást gráfként is modellezhetnénk. Minden járat egy él, és minden elágazás egy csomópont. Az informatikában a gráfok fontos adatstruktúrát jelentenek, melyek valódi rendszerek modellezését segítik, és számos algoritmus alapját képezik.





benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Betűpárok (2013-CA-06)

Az egyforma betűt tartalmazó körök és négyzetek egy párt alkothatnak.

Két szabály van:

- 1) Minden körnek és négyzetnek csak egy párja lehet.
- 2) A vonalak nem keresztezhetik egymást.

Alkoss annyi párt, amennyit csak tudsz! Ügyelj a két szabályra!

Legfeljebb hány pár alkotható?

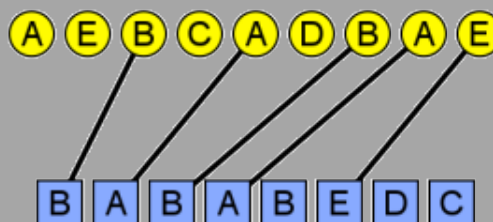
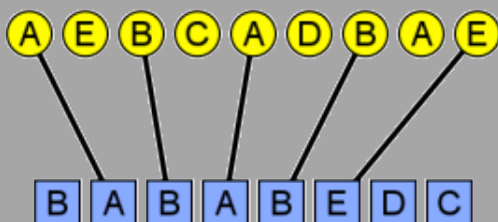
A E B C A D B A E

- A.) 4 pár
B.) 5 pár
C.) 6 pár
D.) 7 pár

B A B A B E D C

„B” válasz a helyes:

Legfeljebb 5 párt lehet létrehozni. Erre két lehetőség van:



5 párnál többet nem lehet összekötni. Az E kört a bal felső sarokban ki lehet zárni, mert a továbbiakban bal oldalon fent csak egy A körrel, jobbra lent pedig csak egy C és egy D négyzettel lehetne párt alkotni. Ez összesen 3 párt jelentene. Ugyanezzel a logikával ki lehet zárni a C és D köröket a felső sorból.

A felső sor C és D körei, ill. a bal felső sarokban lévő E kör kiesése után nem marad sok lehetőség.

Ez informatika!

A fenti és a lenti sor betűiből is hozzuk létre a lehető leghosszabb szót, ugyanazt a szót mindkét sorban (a betűk sorrendjét nem lehet megváltoztatni). Ez a jelenség a leghosszabb közös részsorozat problémájaként ismert. Ezt a módszert alkalmazzák például, amikor két szöveges fájl közti egyezéseket vizsgálnak. Programozók használják, ha közösen dolgoznak egy nagyobb programon, hogy rögtön lássák, hol lett továbbfejlesztve, hol töröltek, változtattak meg vagy adtak hozzá valamit.

A leghosszabb közös részsorozatot a dinamikus programozás segítségével szokták megkeresni. Ennek során egy problémát több lépésben oldanak meg, mindig az egész problémának egy kisebb egységét megoldva.



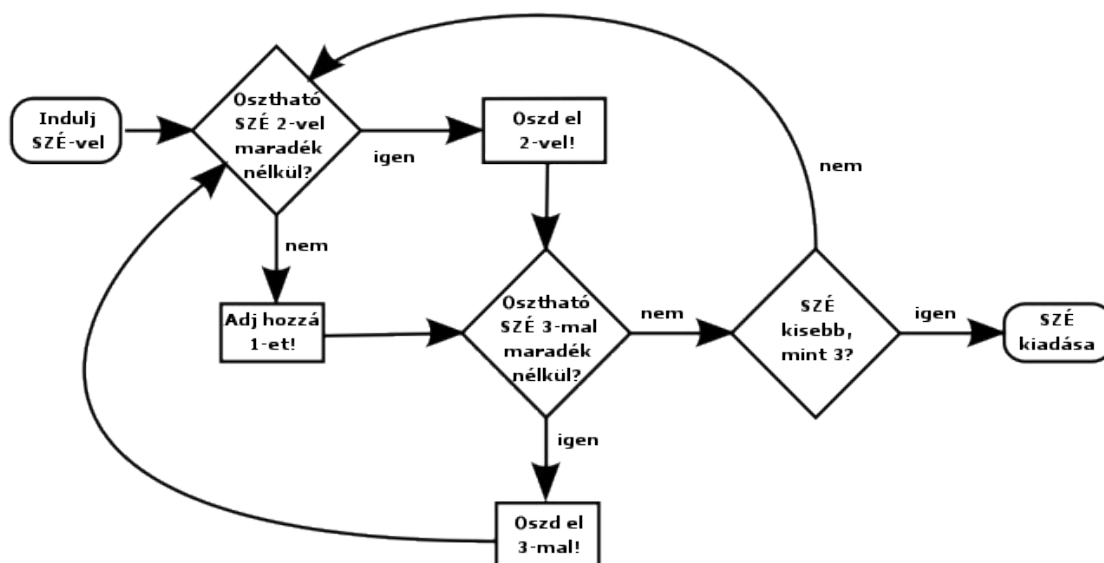
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Folyamatábra (2013-CH-01)

Az iskolában a hódok folyamatábrákról is tanulnak. Ezeken az ábrákon azonban nem víz folyik, hanem egyes műveletek, intézkedések lehetséges következményei kerülnek számbavételre.

Ezen a folyamatábrán egy számértéket (SZÉ) változtatottunk meg különböző műveletekkel. A választási lehetőségek a számérték tulajdonságaira vonatkozó kérdésektől függenek.

Ha a 18-as számmal kezdünk, akkor melyik számérték kerül a végén kiadásra?



A.) 0 B.) 1 C.) 2 D.) 3

„C” válasz a helyes:

Ez a műveleti folyamat:

Kezdd 18-cal

- 18 maradék nélkül osztható 2-vel? Igen. 18 osztva 2-vel 9.
- 9 maradék nélkül osztható 3-mal? Igen. 9 3-mal osztva 3.
- 3 maradék nélkül osztható 2-vel? Nem. Ha a 3-hoz 1-et hozzáadunk, 4-et kapunk.
- 4 maradék nélkül osztható 3-mal? Nem
- 4 kisebb 3-nál? Nem.
- 4 maradék nélkül osztható 2-vel? Igen. 4 osztva 2-vel 2
- 2 kisebb 3-nál? Igen. 2 lesz a megnevezett számérték.

Ez informatika!

Az informatikában a programok fontos részleteinek lefutását jelenítik meg folyamatábrák segítségével. Ezek tulajdonképpen a program válaszai (reakciói) a felhasználó különböző lépéseire. Léteznek folyamatábra-szerű grafikákon alapuló programrendszerek is, pl. <http://scratch.mit.edu/>

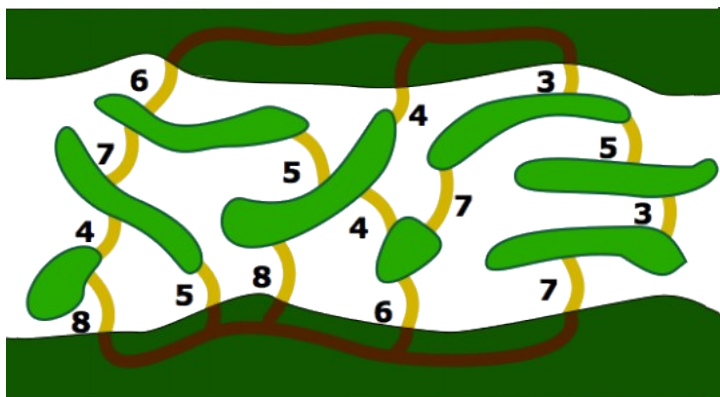


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Hídépítés (2013-CH-02)

A folyóban sok-sok sziget található. Hód Huba hidat akar építeni. Célja a következő: a szigeteket mindkét folyóparttól kiindulva hidakon, illetve part menti utakon keresztül megközelíthetővé kell tenni.

Huba lehetőleg minél kevesebb napot szeretne munkával tölteni. A térképen ezért az összes helyet bejelölte, ahol hidat lehet építeni. A helyek mellett szerepel az is, hogy hány nap szükséges az adott híd megépítéséhez.



Mennyi az a lehető legkevesebb munkanap, mely alatt Huba elérheti a célját!

- A.) 35 munkanap
- B.) 39 munkanap
- C.) 42 munkanap
- D.) 45 munkanap

„B” válasz a helyes:

Ha ebből a tervből bármely hidat eltávolítanánk, legalább egy sziget vagy partszakasz megközelíthetetlen maradna. Minden megépítendő hídhoz legfeljebb annyi munkanapra van szükség, mint a nem megépítendő hidakhoz. Ezen felül a bemutatott megoldásnak a lehető legjobbnak, legkedvezőbbnek kell lennie.

Ez informatika!

Az informatikában az ebben a feladatban felvázolt problémát a „minimális feszítőfa” megkeresésének nevezzük. Ezt hatékonyan az úgynevezett Kruskal-algoritmussal lehet megoldani, mely a következő logikát követi:

1. Kezdjük hidak nélkül.
2. Minden lépésnél azt a hidat építsük meg, amelyikhez a legkevesebb munkanapra van szükségünk, kivéve, ha az adott hídra már nincs szükségünk, hiszen mindkét partja már más hidakon vagy utakon keresztül elérhető.

Természetesen az informatikában ez a probléma nem szigetekkel, hidakkal és megépítésükhöz szükséges munkanapokkal áll összefüggésben. Sokkal inkább gráfokkal, melyek például hálózatok ábrázolása során jönnek létre. Ezek csomópontokból (a mi esetünkben szigetekből) és élekből (a mi esetünkben hidakból) állnak.



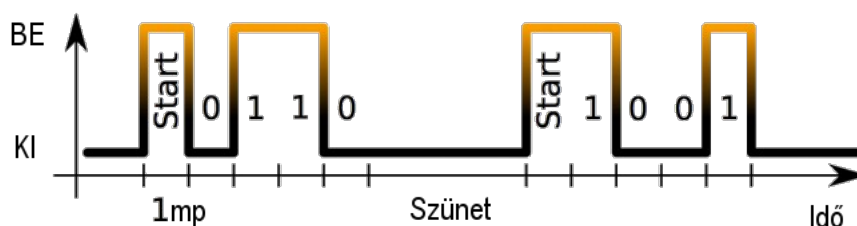
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Sorozatos üzenetek (2013-CH-03)

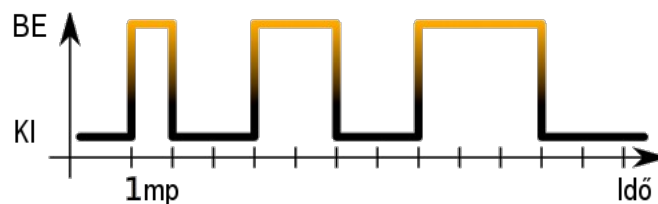
Aliz és Botond éjszakánként a zseblámpájukkal szeretnének üzeneteket közvetíteni.

Négy számjegyből álló blokkokat küldenek. A számjegy 0 vagy 1. Egy számjegyblokk indításához egy másodpercig bekapcsolják a lámpát. Ezután jön a 4 számjegy, másodperces ritmusban. Az égő lámpa 1-et, a kikapcsolt 0-át jelent. A következő blokkig egy legalább egy másodperces szünet következik, kikapcsolt zseblámpával.

A példa a következő számjegyblokkok közvetítését szemlélteti: 0110 és 1001:



Mely számjegyblokkokat közvetítik az alábbi példában?



- A.) 0011 és 1100 számjegyblokkokat
- B.) 1100 és 0011 számjegyblokkokat
- C.) Csak a 0101 számjegyblokkot
- D.) 0011 és 1110 számjegyblokkokat

„A” válasz a helyes:

Az üzenetközvetítés első része 5 másodpercig tart: Kezdmásodperc és 4 számjegy: 0011. Aztán követi egy 2 másodperces szünet. Ezután jön az üzenet második része: kezdmásodperc és 4 számjegy: 1100.

A B válasz pont a fordítottját jelentené, a lámpafény 0-át, a kikapcsolt lámpa 1-et jelentene.

A C választ választók, nem vették figyelembe, hogy minden számjegy csak 1 másodpercig tart.

A D válasznál a második számjegyblokk kezdmásodpercét értelmezték félre.

Ez informatika!

Az adatokat valóban a feladatban leírtak szerint közvetítik, a széles körben elterjedt

RS-232-Protokoll dióhéjban valóban így funkcionál. Habár ezt az eljárást már az 1960-as években feltalálták, a mai napig használatban maradt, hiszen ez a készülékek között egyszerű, megbízható, és mindenekelőtt kompatibilis kommunikációs lehetőséget biztosít. A betűk kódtáblázatok segítségével (pl. ASCII vagy Unicode) számokká alakíthatóak, melyek nullák és egyesek (bitek) sorozatává alakíthatók át. Ma általában egy-egy blokk 8 bitet tartalmaz.

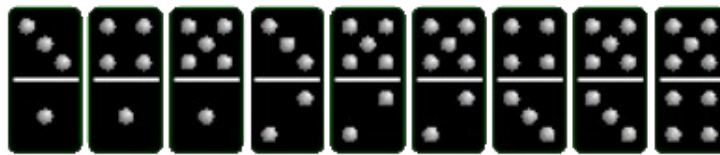


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Dominó (2013-CH-04)

Ebben a feladatban néhány dominót látsz. A dominódaraboknak két oldala van, mindkét oldalukon egy-egy, a dobókockához hasonlatos számokat mutató pontokkal.

A dominókból egy kört tudsz készíteni. Ebben a körben a dominódaraboknak azonos számokat mutató oldalai kell, hogy találkozzanak egymással.

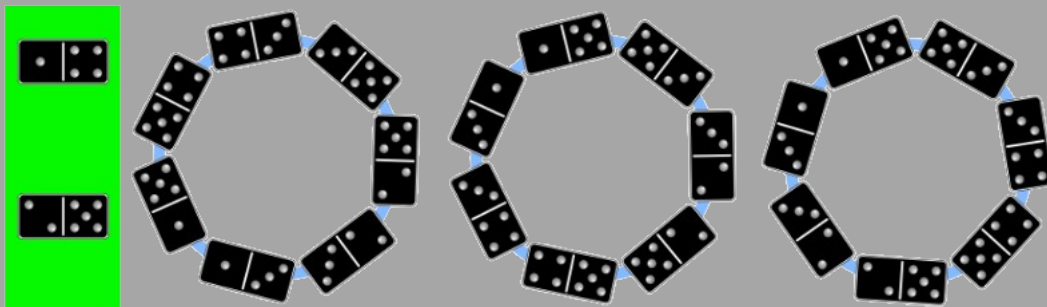


Hány dominó felhasználásával tudod elkészíteni a lehető legnagyobb dominókört?

- A.) 6 dominó
- B.) 7 dominó
- C.) 8 dominó
- D.) 9 dominó

„B” válasz a helyes:

9 dominódarab áll rendelkezésedre. Az 1, 2, 4 és 5 számok páratlanul vannak. Tehát ezek közül a számok közül legalább két dominódarab felesleges marad (a képen balra láthatod). A másik 7 dominót különböző módon rendezheted körbe (a képen 3 példát látsz).



Ez informatika!

Ezt a problémát az informatikában gráfokkal modellezhetjük. A csomópontok a számértékek, az élek pedig a számokhoz csatlakozó dominódaraboknak felelnek meg. A dominókör pedig egy zárt szakasz a gráfon belül, amit egy ceruza felemelése nélkül is meg tudnánk rajzolni. Az ilyen szakaszokat egy híres svájci matematikus után Euler-körnek nevezzük.

Leonhard Euler bizonyította, hogy ha minden csomópontból páros számú él indul ki, akkor Euler-sétát kapunk

<http://hu.wikipedia.org/wiki/Euler-k%C3%B6r>

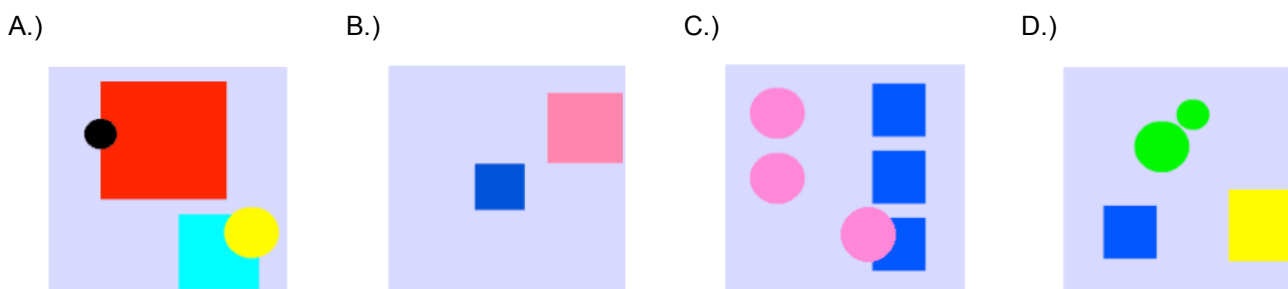
	benjamin	nehéz	közepes	könnyű
	kadét	nehéz	közepes	könnyű
	junior	nehéz	közepes	könnyű
	senior	nehéz	közepes	könnyű

Véletlen képek (2013-DE-02)

Egy gyárban a következő módszerrel készítenek csomagolópapírokat: A nyomtatógép színes köröket és négyszögeket tervez, majd ezeket papírlapokra nyomtatja. A masina minden alkalommal a következő utasításokat hajtja végre:

1. Tervezz egy kört, színezd ki egy véletlenszerűen kiválasztott színnel és nevezd el K-nak
2. Ismételd meg véletlenszerűen sokszor a következő 4 utasításból álló blokkot
 - 2.a Tervezz egy véletlenszerűen kiválasztott színű és nagyságú négyszöget, majd nevezd N-nek
 - 2.b Véletlenszerűen határozd meg K méretét NAGYnak vagy KICSINEK
 - 2.c Nyomtasd K-t egy véletlenszerűen kiválasztott helyre a papírlapon
 - 2.d Nyomtasd N-t egy véletlenszerűen kiválasztott helyre a papírlapon

Melyik papírlapot NEM ezzel a nyomtatógéppel készítették?



„A” válasz a helyes:

Az A papírlapon két különböző színű kör található. A nyomtatógép programja első lépésben kiválaszt egy kört, és a későbbiekben a körnek csupán a méretén változtat. Tehát a papírlapon az összes körnek ugyanolyan színűnek kell lennie. Ezen kívül a gép legalább egy kört nem egy négyszögre nyomtat, hiszen a nyomtatás sorrendje kör – négyszög – kör – négyszög. A program lefutása után a gép ugyanannyi kört nyomtat, mint négyzetet. Ugyanakkor a nyomtató készíthette a B papírlapot is, ha véletlenül a négyzeteket pont a körökre nyomtatta. Az is lehet, hogy a körök véletlenül a háttérrel azonos színt kaptak. A C és D képen ugyanannyi kör és négyzet található. A köröknek mindkét papírlapon azonos színe van, és legfeljebb két különböző méretűek. Ezeket a papírlapokat a nyomtatógép készíthette.

Ez informatika!

Napjainkban igen sok firm használ számítógépek által automatikusan előállított képeket. Ilyenkor a véletlennek fontos szerep jut, ugyanis a tervezők legtöbbször csak absztrakt szabályokat határoznak meg a képek formálásához. Az összetett mesterséges struktúrák apró részleteit (fák ágait és leveleit, állatok bundájának színét és szőrszárait) számítógépek alkotják meg, az alkotás módja nincs meghatározva, de összhangban működik a megadott szabályokkal.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Italautomata (2013-DE-05)

Jaj, ne! Az új italautomatának csak két gombja van: A gomb és B gomb. Pedig négy ital van a választékban: a forróitalok között kávé és tea, a hidegitalok között almale és ásványvíz.



A ravasz gondnok úgy programozta az automatát, hogy a két gomb megnyomásával négy ital közül választhatsz: Nyomd meg először az A gombot a forróitalok, vagy a B gombot hidegitalok választásához. Ezután az A gombbal kávé, vagy a B gombbal teát, illetve az A gombbal almalevet, vagy a B gombbal ásványvizet választhatsz.

Sajnos a gondnok nem akar használati útmutatást kinyomtatni. Így a diákok között különböző útmutatások keringenek az italautomata használatával kapcsolatban. Néhány közülük azonban téves.

Itt egy példa a helyes utasításra: Ha almalevet akarsz inni, nyomd meg először az B gombot, majd az A gombot!

Melyik utasítás helyes?

- A.) Ha két forróitalt akarsz inni, nyomd meg először az A, majd ismét az A gombot!
- B.) Ha meleg teát akarsz inni, nyomd meg először az A gombot, majd a B gombot!
- C.) Ha hideg teát akarsz inni, nyomd meg először az B, majd ismét a B gombot!
- D.) Ha ásványvizet akarsz inni, nyomd meg a B gombot!

„B” válasz a helyes:

Minden italt két gombnyomással választhatunk ki:

- A.) A – A forróital – kávé jelent. Ez csak egy ital.
- B.) A – B forróital – teát jelent. Ez helyes.
- C.) B – B hidegital – ásványvizet jelent. Ennél az automatánál nem lehet hideg teát kapni.
- D.) B hidegitalt jelent. Még egy gombot meg kell nyomni a választáshoz.

Ez informatika!

A feladat a kódolással van összefüggésben. Két gomb esetében kettes kódhosszúságra van szükségünk ahhoz, hogy négy italt bekódoljunk (AA, AB, BA, BB).

Ezen kívül a feladatban véges automaták is megjelennek. Ezek képzeletbeli masinák, melyekkel valódi készülékek viselkedését, mint állapotátmenetek sorozatát, modellezhetjük. Az italautomata irányításának leírásához szükségünk van egy kiinduló állapotra, két állapotra a forró- és hidegitalokhoz, és négy végállapotra.

A kiinduló állapotból az automata csak forró vagy hideg állapotra tud váltani. A forróból csak a kávé vagy a tea, a hidegből csak az almale vagy az ásványvíz érhető el. A diagram segít a kérdés megválaszolásában, könnyen leolvasható, hogy az A-A a kávéhoz, B-B az ásványvízhez vezet, B csupán a forróital állapotot jelenti, nem vezet tovább.



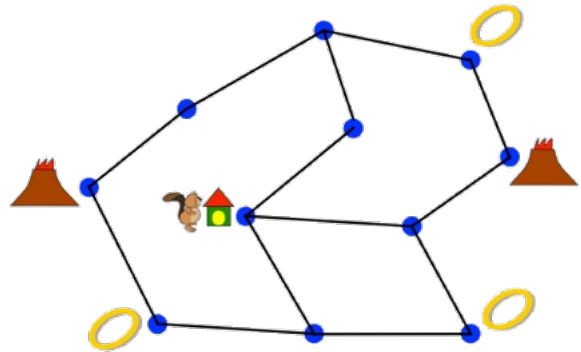
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Hobbit Hód (2013-FR-02)

FrodHód utazásra készül. 3 gyűrűt kell összeszednie és egy vulkánba hajítania, csak ezután térhet vissza házikójába.

FrodHódnak van egy térképe, mely minden utat két pont közötti szakaszként ábrázol. Egy útszakasz bejárásához 1 napra van szükség.

FrodHód egy útszakaszon többször is áthaladhat, és nem kell minden útszakaszt bejárnia.

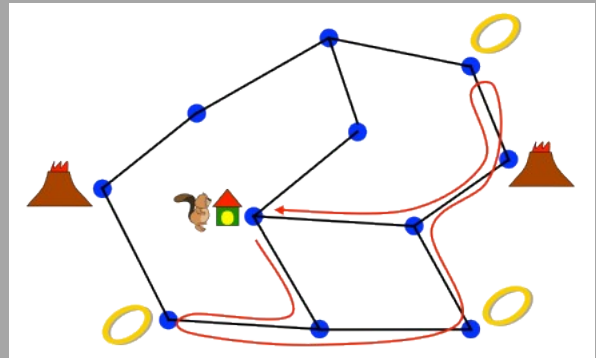


Legkevesebb hány napra van szüksége FrodHódnak a nagy utazáshoz?

- A.) 9 napra
- B.) 10 napra
- C.) 11 napra
- D.) 12 napra

„B” válasz a helyes:

2 napi gyaloglás után szerzi meg HobbitHód az első gyűrűt. A 4. nap végére a másodikat. A 7. nap végére mind a három gyűrűt megszerzi. A 8. napon a jobb oldali vulkánba dobja őket, és a 10. napon hazatért. Rövidebb út nincs.



Ez informatika!

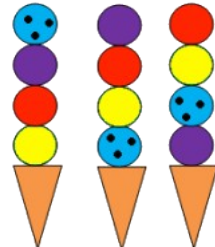
Itt egy gráfon belül egy olyan utat kerestünk, mely jó néhány feltételnek szükségszerűen megfelel. Ha próbálgatással és gondolkodással, vagy csak egészen véletlenül rátalálunk egy útra, nehéz eldönteni, hogy az vajon valóban a legrövidebb lehetséges útvonal-e. Az informatika ezért különböző célokra nagyszámú szisztematikus módszert fejlesztett ki, melyek segítségével az optimális megoldás megtalálható. Azonban az is előfordulhat, hogy az optimális megoldás keresése drágább lenne, mintha egy könnyen megtalálható, ugyanakkor nem teljesen optimális megoldást választanánk.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

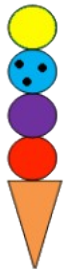
Fagyiautomata (2013-HU-01)

Egy különleges fagyiautomata négygombócos adagokat ad, még hozzá nem véletlenszerűen. A képen jobbról balra látható a legutóbbi 3 tölcsér, amit a gép készített.

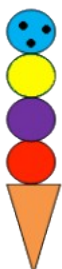


Melyik tölcsér lesz a következő?

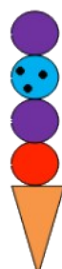
A.)



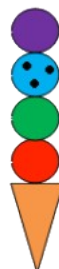
B.)



C.)



D.)



„A” válasz a helyes:

A fagyiautomata mindig ugyanabból a négy fajta fagyiból állít össze egy adagot. A legfelső gombóc a következő tölcsérben legalulra kerül. A másik három gombóc sorrendje változatlan marad.

Ez informatika!

Ha egy önműködő gép (automata) működési elvére próbálunk rájönni, akkor valójában az azt működtető program működési elvét igyekszünk megfejteni. Ebben fontos szerepet játszanak az ismétlődő folyamatok, az úgynevezett ciklusok. A működés közben megfigyelt mintákból lehet következtetni az azok alapjául szolgáló algoritmusra.



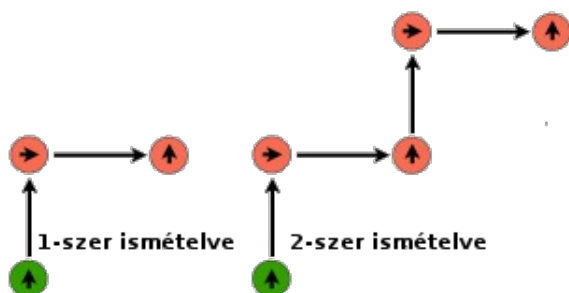
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Úszás (2013-HU-04)

Hódunk érdekes úszási stílust szeretne kifejleszteni.

Minden úszási stílus farokcsapások és fordulások sorozatából áll. Ezek ismétlődnek újra és újra.

Például ha a "farok csapás - jobbra fordulás - farok csapás - balra fordulás" sorozatot hajtja végre, akkor a képen látható utat ússza be.



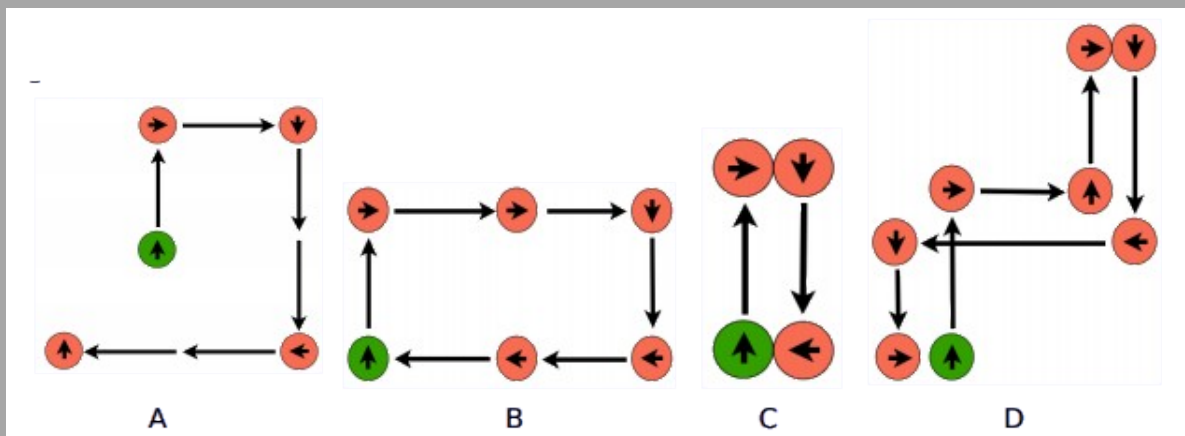
De nem minden ilyen ismétléssel tud eljutni valahova.

Az alábbi sorozatok melyikének ismétlésével jut a legtávolabb a kiindulási pontjától?

- A.) farok csapás - jobbra fordulás - farok csapás - jobbra fordulás - farok csapás - farok csapás - jobbra fordulás - farok csapás - farok csapás - jobbra fordulás
- B.) farok csapás - jobbra fordulás - farok csapás - farok csapás - jobbra fordulás - farok csapás - jobbra fordulás - farok csapás - farok csapás - jobbra fordulás
- C.) farok csapás - jobbra fordulás - jobbra fordulás - farok csapás - jobbra fordulás - jobbra fordulás
- D.) farok csapás - jobbra fordulás - farok csapás - balra fordulás - farok csapás - jobbra fordulás - jobbra fordulás - farok csapás - jobbra fordulás - farok csapás - balra fordulás - farok csapás - balra fordulás - balra fordulás

„A” válasz a helyes:

Az egyes úszásmódok eredményeit az alábbi ábra mutatja:



Ez informatika!

A feladat a procedurális programozás, az algoritmikus gondolkodás példája, melyet a logo (Papert, MIT, ...) programozási nyelvvel élvezhetünk a leginkább.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Akciós? (2013-HU-07)

A Hódboltban szeretnék, ha bizonyos árukat párban vásárolnának meg a vevők. A megfelelő párok azonosításához az 1 és 0 számjegyekből álló speciális kódokat használnak.

Két cikk párban van, ha a számaikban szereplő számjegyek különbözőek: azaz ha az egyik számban egy helyen 1 áll, akkor a másikban ugyanott 0, illetve fordítva.

Például a 0110 kóddal rendelkező cikk párja az 1001 kóddal rendelkező.

A következő párok közül melyik NEM megfelelő az akcióban?

- A.) 1010 – 0101
- B.) 0011 – 1100
- C.) 1011 – 0100
- D.) 1101 – 0001

„D” válasz a helyes:

Az 1101 párja a 0010 lenne, illetve a 0001 párja az 1110.

Ez informatika!

A bináris számjegyek, a kettes számrendszer az informatika alapját képezi. Ebben a feladatban a választható párok számai egymás egyes komplementjei, mely a gépi számábrázolás egyik lehetősége a negatív számok ábrázolására (negációs kód) is.

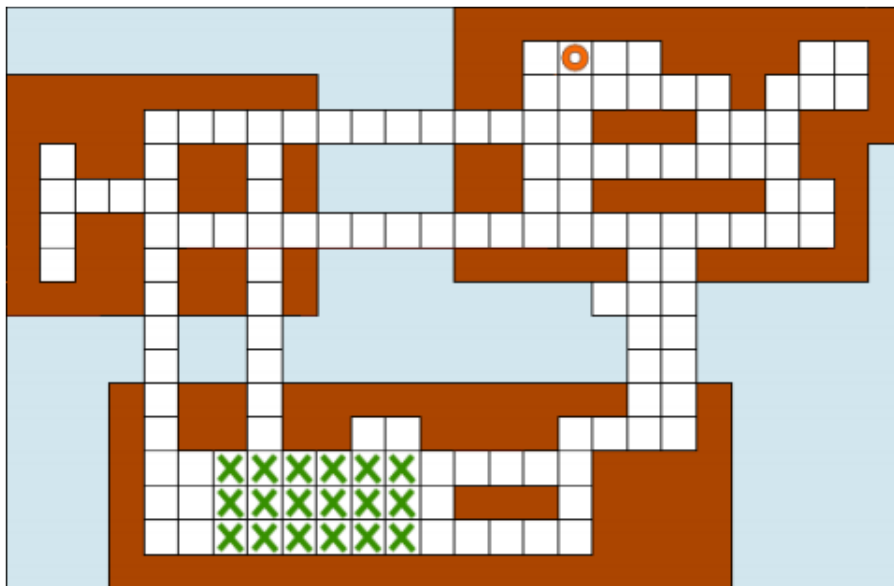
http://hu.wikipedia.org/wiki/Gépi_számaábrázolás



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Hazaút (2013-IT-07)

Igodot szigeteit négyzet alakú parcellákból álló utak hálózák be és hidak kötik össze egymással. Igodot egy perc alatt ér át egyik parcellából a másikba. Csak derékszögben tud fordulni és soha nem haladhat átlósan. Igodot éppen a mezőn van valahol (a keresztekkel jelölt terület), amikor hazatelefonál, hogy a legrövidebb úton elindul hazafelé (a gyűrűvel jelölt terület).



Mennyi idő alatt ér haza Igodot?

- A.) legalább 20 és legfeljebb 24 perc alatt
- B.) pontosan 20 perc alatt
- C.) legalább 21 perc és legfeljebb 25 perc alatt
- D.) pontosan 25 perc alatt

„C” válasz a helyes:

Igodotnak legalább 21 és legfeljebb 25 parcellán kell áthaladnia. Azok a válaszlehetőségek, amelyek pontosan megadják az időt, eleve kiesnek, mert nem mindegyik parcella van ugyanolyan messze a háztól.

Ez informatika!

Sok mindent nem tudunk pontosan, sok kérdésben bizonytalanok vagyunk. Sokszor csak részinformációink vannak. Hogy lehet leírni formálisan az ilyen hiányos tudást, hogy azt számítógépen tárolni, azzal számolni és tervezni lehessen?

Az informatika legkülönbözőbb részterületei dolgoznak ezen a problémán: az „intervallum-aritmetika”, a „statisztika” és a „mintafelismerés” éppúgy, mint az „elmosódott halmazok logikája”, a „mesterséges intelligencia” vagy az „ismeretelmélet”.

Azt, hogy mennyit kell várnunk Igodotra, ki tudjuk számolni. Vagy mégsem? Mi van, ha gondol egyet és tesz egy kerülőt? Mi van, ha útközben megáll beszélgetni? Mi van, ha egyáltalán nem jön haza? Mi van, ha egyáltalán nem is létezik?



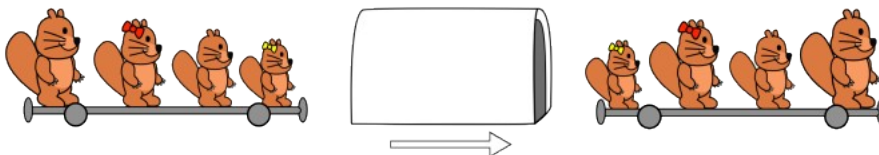
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Varázslatos alagút (2013-JP-02)

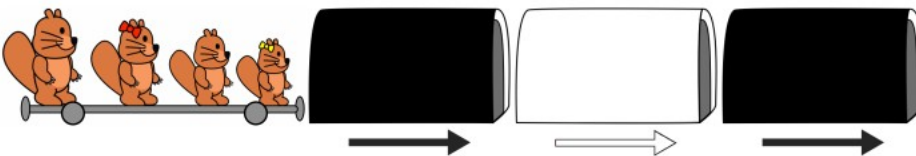
A hódvasút kétféle alagutat használ. Ha a szerelvény egy fekete alagúton megy keresztül, az utasok fordított sorrendben jönnek ki a túloldalon.



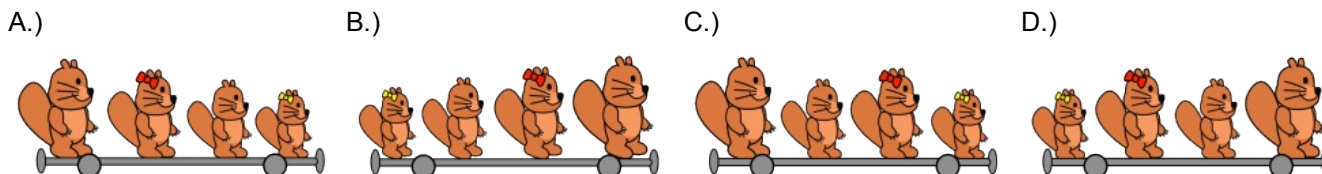
Ha a szerelvény egy fehér alagúton megy keresztül, az első és a hátsó utas helyet cserél.



A mi szerelvényünk három alagúton megy keresztül:



Milyen sorrendben jönnek ki az utasok az utolsó alagútból?

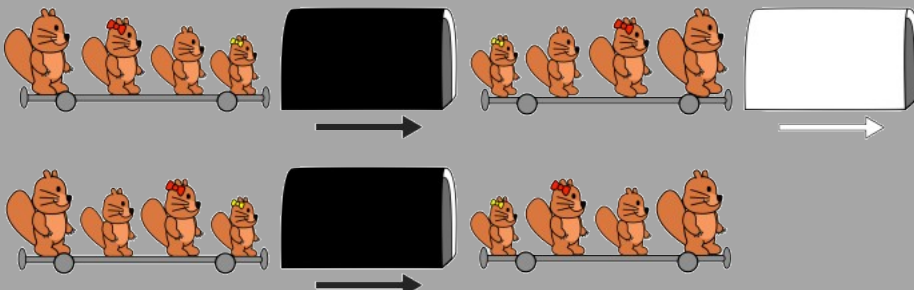
**„D” válasz a helyes:**

A sorrend kezdetben 1-2-3-4.
Az első fekete alagút után 4-3-2-1. A fehér alagút után 1-3-2-4. A második fekete alagút után 4-2-3-1.

Ez informatika!

A fehér és a fekete alagút egy-egy függvényt jelenítenek meg.

Mindkettő egy sorozat elemeinek (a négy hódnak) a sorrendjét változtatja meg. Mindkét „alagút-függvénynek” van egy különleges tulajdonsága: egyúttal saját maguk fordítottjai is. Ha egy kocs két fekete alagúton halad keresztül, a hódok újra a kezdeti sorrendben ülnek rajta. Ugyanez vonatkozik két fehér alagútra is. Egy sok alagútból álló sorozatnál csak azt kell megvizsgálni, hogy páros vagy páratlan számú fehér, ill. fekete alagutat tartalmaz. Így egy sokkal rövidebb alagútsorhoz jutunk, amely ugyanazokat a változásokat eredményezi. 33 fehér és 67 fekete alagút például funkcióját tekintve egy fehér és egy fekete alagútnak felel meg.



Tanároknak: adjuk fel a feladatot 100 kocsival. Először azok fognak jelentkezni – méghozzá a helyes megoldással – akik analitikusan oldják meg a feladatot. Csak később azok, akik végigpróbálgatták, ráadásul a 75 százalékuk valószínűleg rossz megoldással.

Még egy klasszikus 1973-ból az algoritmusok, adatrendszerek és kocsik összefüggéséről:

<http://www.cs.utexas.edu/users/EWD/ewd03xx/EWD365.PDF>

<http://www.cs.utexas.edu/users/EWD/>



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Evezősverseny (2013-JP-04)

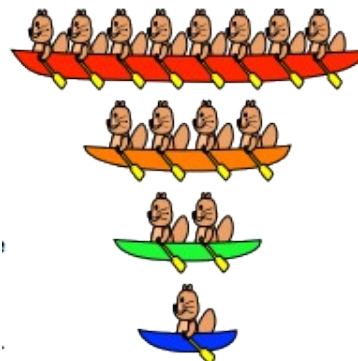
Néhány hód egy evezősversenyen szeretne elindulni. Összesen négy hajójuk van, minden hajóosztályban egy: egy 8 személyes, egy 4 személyes, egy 2 személyes és egy 1 személyes.

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 kezdi, 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:

$$8+4+0+1=13 \text{ hód}$$

„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 ($8+4+2+0=14$).

Ez 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:

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



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Jelzőtűz (2013-JP-07)

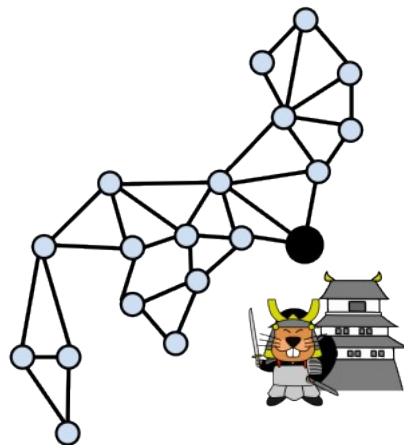
Régen a szamurájok kiépítettek Japánban egy több jelzőállomásból álló hálózatot. Vészhelyzetben jelzőtüzeket gyújtottak az állomásokon, így az egész országot riadóztatni tudták.

A képen a körök jelölik a jelzőállomásokat. A vonallal összekötött állomások szomszédosak.

Ha az egyik állomáson tüzet gyújtanak, ezt a szomszédos állomásokon egy perc után veszik észre, és azonnal meg is gyújtják a saját jelzőtüzüket.

További egy perc elteltével tehát a szomszédok szomszédai is meggyújtják a jelzőtüzet. Ez így megy tovább, amíg minden állomáson ki nem gyullad a jelzőtűz.

Egy nap meggyújtják a főhadiszálláson lévő állomás (a nagyobb, fekete kör) jelzőtüzet.



Hány perc után gyullad ki a jelzőtűz az összes állomáson?

- A.) 4 perc után
- B.) 5 perc után
- C.) 6 perc után
- D.) 8 perc után

„B” válasz a helyes:

Az összes jelzőállomáson meggyújtják a jelzőtüzet, ahol egy perc után észreveszik a főhadiszállás jelzőtüzet. Azokon az állomásokon, ahol a második perc elteltével észreveszik ezeket a jelzőtüzeket, szintén tüzet gyújtanak. Azokon az állomásokon, ahol a harmadik perc elteltével észreveszik a legutóbb gyújtott tüzeket, szintén meggyújtják a jelzőtüzet. És így tovább.

Ez informatika!

A jelzőállomásokat és a köztük lévő kapcsolatokat – hogy látótávolságban vannak egymástól – egy gráf segítségével ábrázoltuk. A gráfot mint adatrendszert gyakran használják az informatikában egy térkép információinak sematizált ábrázolására. Kitűnő példa az ilyen sematikus ábrára egy vagy több metróvonal térképe.

A fent leírt esetben az időbeli távolság a szomszédos állomások között megegyezik. A megoldás, hogy minden állomástól kiszámoljuk a legrövidebb utat a főhadiszállásig, majd ezek közül kiválasztjuk a leghosszabbat. Ez megoldható például egy „szélességi keresés” segítségével.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

RAID (2013-LV-02)

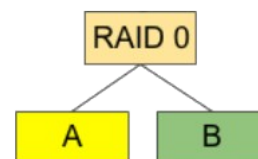
A RAID egy olyan technológia, amikor több merevlemezt kapcsolnak össze a közösen szervezett adattárolásért.

Többek között a RAID-nek ez a két típusa létezik:

RAID0

Az adatok csak az egyik RAID-be csatlakoztatott merevlemezen tárolódnak. Az egyes merevlemezek tartalma mind különböző. Ezért az adatbiztonság nem magasabb, mint egyetlen lemez esetében.

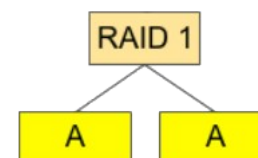
A kép egy ilyen RAID-et (RAID0) ábrázol két merevlemezrel.



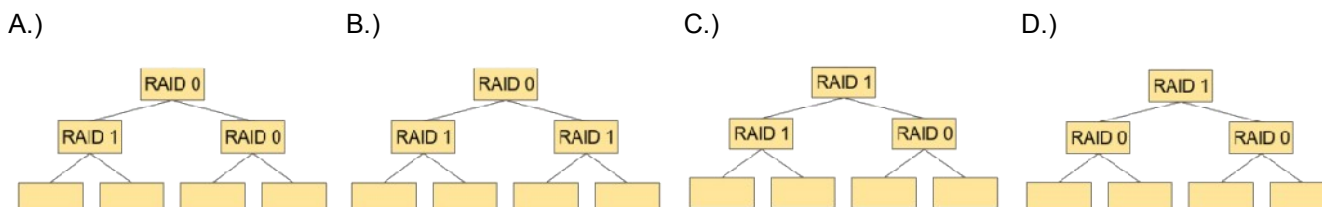
RAID1

Az adatok több merevlemezen kerülnek tárolásra úgy, hogy a merevlemezek tartalma mindig azonos. Így kevesebb adat tárolható, de az adatbiztonság annál magasabb, minél több merevlemezen tárolja ugyanazt az adatot.

A kép egy ilyen RAID-et (RAID1) ábrázol két merevlemezrel.



Az alábbi RAID-ek közül melyik az, amelyiken NEM történik adatvesztés akkor sem, ha két tetszőleges merevlemez tönkremegy?



„C” válasz a helyes:

A C esetben az adatok három merevlemezen kerültek tárolásra, kétszer a Raid1-ben (balra lent) és egyszer a Raid0-ban (jobbra lent). Amennyiben ezek közül tetszőleges kettőt tönkremegy, még mindig marad egy másolatunk.

A és B esetben adatokat veszítünk, ha a bal alsó Raid1-ben megy tönkre mindkét merevlemez, hiszen a jobb oldali Raid-ekben nincs róluk másolat. A D esetben akkor veszítünk adatot, ha a két tönkremenő merevlemezről egy a jobb, egy meg a bal alsó Raid0 alatt található.

Ez informatika!

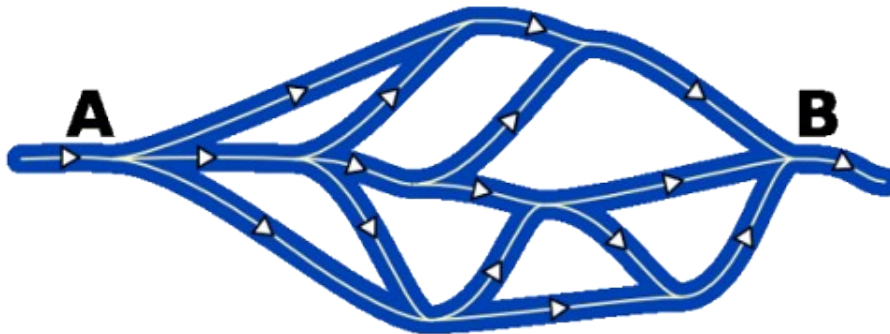
A bemutatott RAID technológiákkal egyrészt az adatbiztonságot növelhetjük (RAID1), másrészt a tárolt adatok gyorsabb elérhetőségét biztosíthatjuk. RAID-rendszert mint szoftveresen (Software-RAID, mind hardveresen (RAID-Controller) megvalósíthatunk.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Folyóellenőrzés (2013-NL-02)

A hódok rendszeresen szerveznek közös folyóellenőrzést. Ehhez a folyónak minden ágán legalább egy hódnak végig kell úsznia. Egyszerre indulnak A pontból és B pontban találkoznak újra. A hódok csak egy irányba úszhatnak az áramlással.



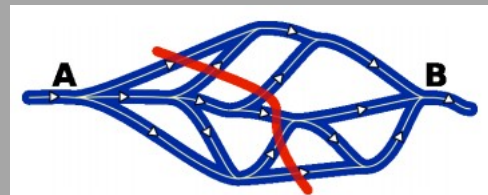
Legalább hány hódra van szükség a közös folyóellenőrzéshez?

- A.) 3 hódra
- B.) 4 hódra
- C.) 5 hódra
- D.) 6 hódra

„D” válasz a helyes:

Legalább 6 hódra van szükség. Egyikük a jobb szélső, másikuk a bal szélső ágon úszik végig. A harmadik középen, majd balra tart. A negyedik kétszer középre tart, majd balra. Az ötödik kétszer középre, majd jobbra, végül balra tart. A hatodik középre, jobbra, balra és megint jobbra tart.

A piros vonal hat folyóágot érint. Lehetetlen, hogy egy hód A pontból B pontba tartva ezek közül egynél több ágon áthaladjon. Másrészt nem lehet olyan vonalat húzni, amelyik hatnál több folyóágot keresztez. Hat hód tehát elegendő a feladatra.



Ez informatika!

A folyóágakból álló rendszert egy irányított gráffal lehet ábrázolni, ahol az elágazások a csomópontoknak, a folyóágak az éleknek, a piros vonal pedig a gráf maximális vágásának [maximum cut] felel meg.

A legtöbb folyóproblémára az informatikának van hatékony megoldási algoritmus, amelyeket logisztikai és kommunikációs rendszerek tervezésénél és optimalizálásánál sokféleképpen fel lehet használni.

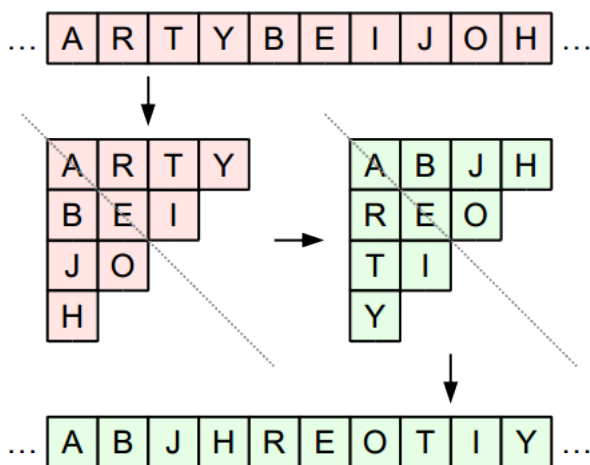


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Háromszög kód (2013-NL-06)

Betti titkos üzenetet szeretne küldeni a legjobb barátnőjének. Először kitörli a szövegből az összes szóközt, majd a megmaradt szöveg kódolására a következő eljárást próbálja ki:

- A szöveget részekre bontja, melyek 10 karakter (betű, írásjel) hosszúak.
- Minden szövegdarabot háromszög formájában ír föl (lásd a képen)
- A háromszöget egy átlós tengely mentén tükrözi (lásd a képen)
- A háromszöget végül újra szövegrészletként írja le (lásd a képen)



Betti legjobb barátnője a következő kódolt üzenetet kapja: MEK?INÉNOD

Hogy hangzik az eredeti kérdés?

- A.) MINDENOKÉ?
- B.) OKMINDEN?
- C.) MINDENKOR?
- D.) MONDDEL?

„A” válasz a helyes:

Ezt könnyen megfejthetjük, ha visszafelé elvégezzük a kódolás folyamatát.

Akár a kódolás műveleteinek sorrendjében is elvégezhetjük a dekódolást, hiszen az eljárás szimmetrikus.

Ez informatika!

Ez az egyszerű kódolási eljárás tulajdonképpen a szkütalé egy változata, melyet már 2500 évvel ezelőtt is alkalmaztak. Az ehhez hasonló kódolási módszereket könnyen meg lehet fejteni, főleg ha az átkódolt szöveg egész hosszú. A klasszikus titkosírásokkal ellentétben- mint például a Cézár-kódolás, vagy a Vigenère-kódolás, a szkütalé gyorsan, különösebb erőfeszítések nélkül alkalmazható. Az informatika sokkal biztonságosabb eljárásokat ajánl Bettinek, például a One-Time-Pad-et. Ehhez csak papírra és ceruzára van szükség.

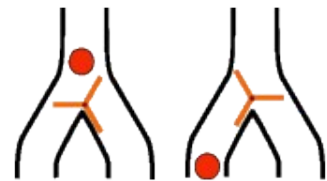


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

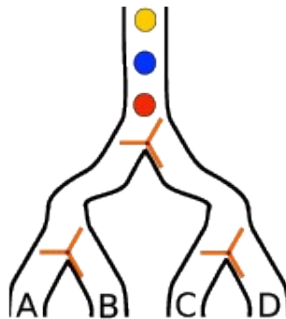
Flipflop (2013-RU-05)

A flipflop egy olyan szerkezet, aminek két állása van. Amint kap egy jelet, átáll a másik állásba. A hódok flipflop-rendszere a következőképpen működik.

A labda (a jel) felülről érkezik, és a flipflop állásától függően két irányba eshet tovább, jobbra vagy balra. Eközben úgy forgatja el a flipflopot, hogy a következő labda a másik irányba fog esni.



A hód épített magának egy szerkezetet flipflopokkal, amelyik így néz ki:



Melyik csőből fog a harmadik (sárga) labda kiesni?

- A.) A csőből
- B.) B csőből
- C.) C csőből
- D.) D csőből

„B” válasz a helyes:

Az első (piros) labda az első flipflopnál balra, a bal alsó flipflopnál megint balra fog esni, az A csővön. A második (kék) labda a felső flipflopnál jobbra, a jobb alsó flipflopnál balra fog esni, a C csővön. A harmadik (sárga) labda a felső flipflopnál balra, a bal alsó flipflopnál jobbra fog esni, a B csővön.

Ez informatika!

Mivel egy flipflopnak csak két lehetséges állása van, és ezekből egyszerre mindig csak egy pozíciót vehet fel, tökéletesen alkalmas egy bit tárolására.

A bit a legkisebb információs egység. Egy bit csupán két értéket vehet fel: „igen” vagy „nem”, „1” vagy „0”, „plusz” vagy „mínusz”, „jobb” vagy „bal”, stb...

A számítógépekben ezek a kis tárolóegységek („flipflopok”) rendszerint apró áramkörök, amelyekből több milliárd található egy chip-en.



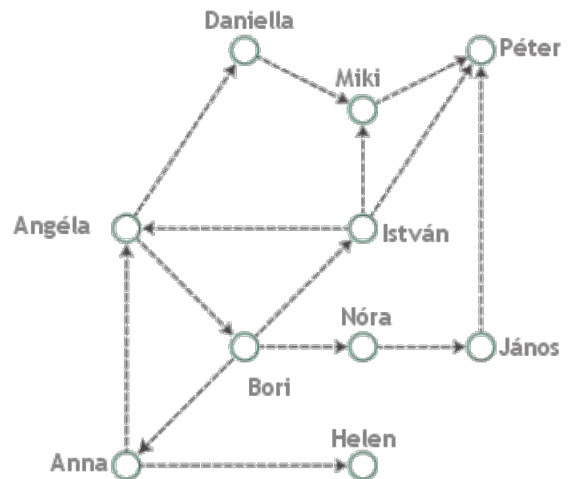
benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Mi újság? (2013-SI-01)

A hódok szeretnek beszélgetni az iskolában és szívesen osztják meg egymással a legújabb híreket.

A kép azt mutatja, hogy ki kinek adja tovább a híreket. István például mindent elmesél a barátainak, Angélának, Mikinek és Péternek.

Ma Anna új hírrel érkezett az iskolába, amelyik gyorsan el is terjedt. A szünetben Helen, Péter, István és János találkoztak és megállapították, hogy Helen és Péter hallotta ugyan az új hírt, István és János azonban nem. Nyilvánvalóan hiányzott valaki az iskolából, és ez akadályozta a hír megszokott terjedését.



Ki hiányzott ma az iskolából?

- A.) Nóra
- B.) Bori
- C.) Angéla
- D.) Miki

„B” válasz a helyes:

Ha Péter hallott Anna új híreről, akkor Angéla, Daniella és Miki sem hiányozhatott. Az, hogy István nem hallotta a hírt, azt jelenti, hogy Ben nem mesélte el neki. Angéla viszont volt iskolában, így elújságozhatta volna Borinak a hírt, ha bent lett volna az iskolában. Tehát Bori hiányzott ma.

„A” válasz helytelen, mert ha Nóra hiányzik, István akkor is értesül a hírről Boritól.

„C” válasz helytelen, mert ha Angéla hiányzik, Péter nem hallotta volt a hírt.

„D” válasz helytelen, mert ha Miki hiányzik, akkor is eljutott volna mindenki máshoz a hír.

Ez informatika!

A hódok, akik az új híreket mesélik egymásnak, a routereknek felelnek meg egy redundáns kapcsolatokkal rendelkező számítógép-hálózatban. Ha valahol megakad a továbbítás, meg kell keresni a hibás routert.

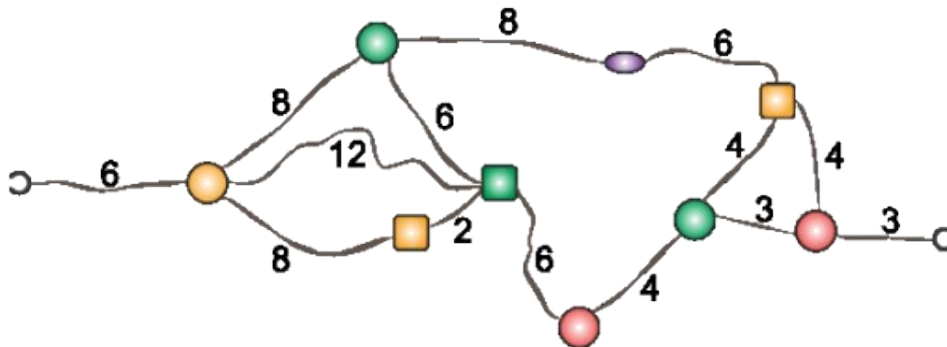


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

A megfelelő nyaklánc (2013-SI-04)

Kim egy nyakláncot fűzött színes gyöngyökből. Hogy illik-e a nyakára?

A számok a gyöngyök közötti madzag hosszát adják meg centiméterben. A jobb és a bal oldalon vannak a kapcsok.



Legfeljebb mekkora lehet Kim nyakának kerülete, hogy a lánc még jó legyen rá?

- A.) 26 cm
- B.) 32 cm
- C.) 34 cm
- D.) 35 cm

„B” válasz a helyes:

A nyaklánc pontosan olyan hosszú, mint a madzag legrövidebb szakasza a két kapocs között:
 $6+8+2+6+4+3+3=32$

Ez informatika!

Gyakorlott informatikusok első pillantásra rájönnek, hogy a nyaklánc hosszúságára vonatkozó kérdést egy másik, kiszámolható kérdéssel lehet helyettesíteni. Ha a nyakláncot mint gráfot vizsgáljuk, ahol a gyöngyök és a kapcsok a csomópontok, a zsinórok pedig az élek, akkor a nyaklánc hossza egyenlő a jobb és bal oldali kapocs közötti legrövidebb távolsággal.

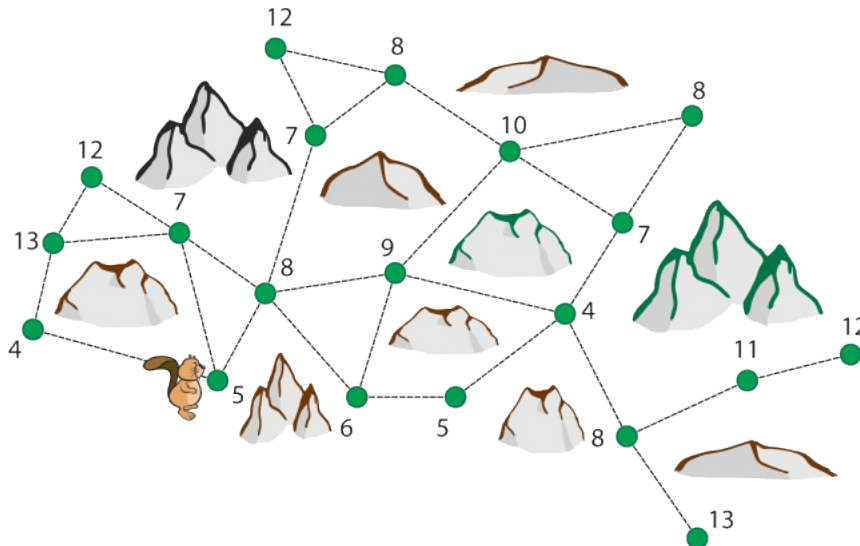
Az informatika hatékony algoritmusokat ismer egy gráf két csomópontja közötti legrövidebb út kiszámolására. A hatékonyság ebben az esetben fontos. A 11 csomópontból és 14 élből álló nyaklánc esetében még éppen ki lehet számolni fejben a megoldást. De már a leggyorsabb busz- illetve vonatút kiszámolása is egy közepesen nagy városban a számtalan csatlakozási lehetőség miatt jóval problémásabb.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

A legmagasabb fa (2013-SI-05)

A térképen néhány fa ponttal van jelölve. Mellette meg van adva a magassága. A hód nem lát messzire a sziklák miatt. Ha az egyik fáról mégis rálát egy másikra, a két pontot vonal köti össze.



A hód a lehető legmagasabb fát szeretné megtalálni és kivágni. Annál a fánál kezdi a keresést, amelyiknek a magassága 5 méter. Innen látja azokat a fákat, amelyeknek a magassága 4, 7 és 8 méter. Mindig a látótávolságon belül lévő legmagasabb fához megy. A második fa tehát a 8 méter magas fa lesz. Ezzel a módszerrel keres egyre magasabb fákat. A keresés akkor ér véget, ha az a fa, amelyiken ül, magasabb, mint bármelyik látótávolságon belül lévő fa. Ezt a fát fogja kivágni.

Milyen magas az a fa, amelyiket a hód végül kivágja?

- A.) 9 méter B.) 10 méter C.) 12 méter D.) 13 méter

„B” válasz a helyes:

A hód az első fáról a 4, a 7 és a 8 méter magas fákat látja. A 8 méteres fa a legmagasabb, magasabb annál is, amelyiken ő ül. Tehát ehhez a fához megy.

Innen két 7, egy 9 és egy 6 méter magas fát lát. Tehát a 9 métereshez megy.

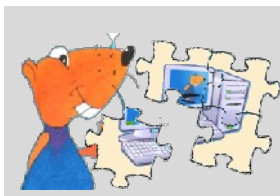
Innen egy 6, egy 8, egy 10 és egy 4 méter magas fát lát. Tehát a 10 métereshez megy.

Innen egy 9, két 8 és egy 7 méteres fát lát. Mindegyik alacsonyabb 10 méternél, tehát a 10 méteres fa a legmagasabb, amit találhatott.

A 11, 12 és 13 méter magas fák ezzel a keresési módszerrel egyszer sem kerülnek látótávolságon belülre

Ez informatika!

Azt a keresési algoritmust, amit itt használtunk, „helyi keresésnek” hívják. Nem egy átfogó, hanem csak egy egészen szűk környezetre, jelen esetben a látható fákra korlátozott keresésről van szó. A feladat azonban rámutat a helyi keresés gyengéjére is: elképzelhető, hogy a globális optimumot – ebben az esetben a legmagasabb fát – nem találjuk meg.

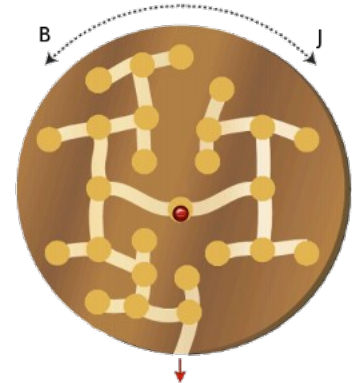


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Ügyességi játék (2012-SI-09)

Kukacok egy fészkekből és összekötőjáratokból álló hálózatot vájtak egy fakorongba. A hódok ügyességi játékot csináltak belőle.

A korong közepén lévő fészkekbe egy piros üveggolyót tesznek. Ha a fakorongot ügyesen forgatják balra (B) és jobbra (J), az üveggolyót az összekötőjáratokon keresztül egy szomszédos fészkekbe tudják juttatni. A cél az, hogy a forgatások segítségével, fészkekről fészkekre haladva eljusson az üveggolyó a kijáratig.



Milyen forgatásokra van szükség, hogy az üveggolyó eljusson a kijáratig?

- A.) B J J B J J
- B.) J J J B
- C.) B J J B J B
- D.) B B B J J

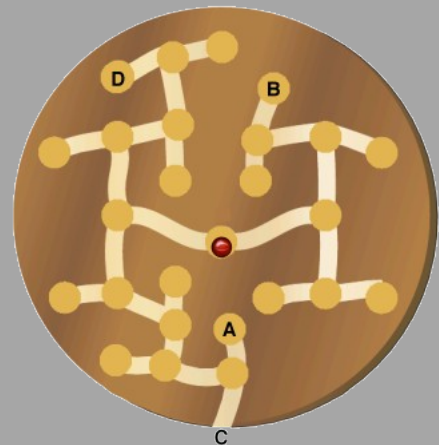
„C” válasz a helyes:

A kép azt mutatja, hogy hova jut a golyó a válaszlehetőségekben szereplő forgatások után.

Időt spórolunk meg, ha észrevevesszük, hogy a cél eléréséhez összesen 6 forgatásra van szükség. Így a „B” és „D” válaszlehetőségeket ki lehet zárni.

Ez informatika!

A kukacok fészkekből („csomópontok”) és összekötőjáratokból („élek”) álló rendszere, ha adatrendszerként értelmezzük, egy „bináris fa”. Gyökere a közepén lévő fészkek. A forgatások „utakat” hoznak létre az egyes „levelekhez”. Ezek közül a „levelek” közül az egyik a kijárat.

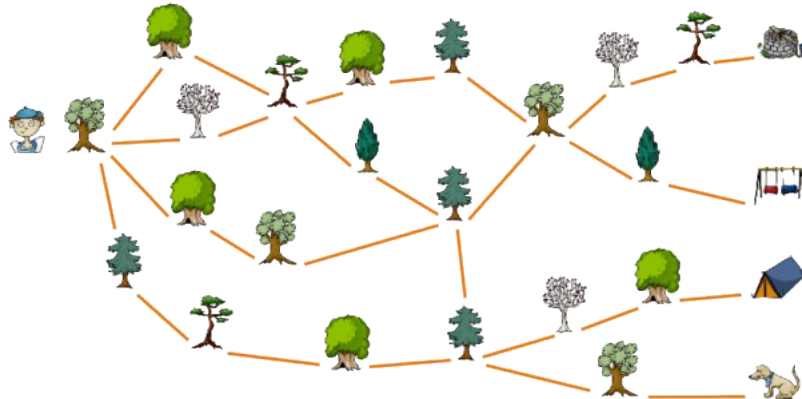




benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Az erdőben (2013-SK-01)

Rupert egy erdőn megy keresztül. Az út végén a kutyája várja.



Milyen sorrendben halad el a fák mellett (balról jobbra)?



„C” válasz a helyes:

Nem szükséges balról jobbra, fáról fára haladva leellenőrizni Rupert útját. Az utolsó két fa segítségével kizárásos alapon rá lehet jönni a helyes megoldásra. Az „A” válaszban az utolsó fa nem stimmel. A „B” és „D” válaszokban az utolsó előtti fa nem stimmel.

Ez informatika!

Az erdő térképe egy gráf. A gráfok mint adatrendszerek fontos szerepet játszanak az informatikában. Itt a fák, a kút, a hinta, a sátor és a kutya jelölik a „csomópontokat”, az útszakaszok pedig az „íveket”.

Egy problémát „visszafelé” vizsgálni érdekes stratégia, amely az informatikában megdöbbentően elegáns megoldásokhoz vezethet.

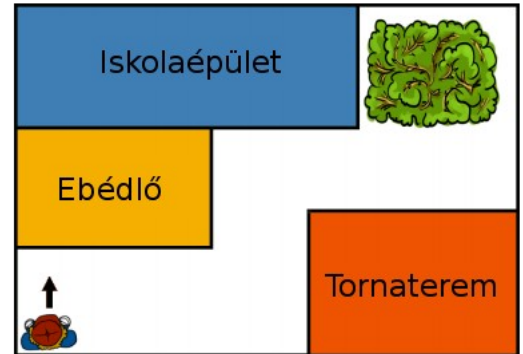


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Irány a bokorba! (2013-SK-04)

A gyerekek robotosát játszanak. Bence játssza a robotot, és csak erre a három parancsra hallgat: `Előre!` `Balra!` és `Jobbra!` Ha a gyerekek azt kiáltják, hogy `Előre!`, Bence előre felé halad, amíg akadályba nem ütközik. Aztán a gyerekek `Balra!`-t kiáltanak, Bence pedig balra fordul. Ha azt kiáltják `Jobbra!`, Bence jobbra fordul.

Bence az iskolaudvar egyik sarkában áll, ahogy azt a képen felülnézetből láthatod. A fiú a műhely felé néz. A gyerekek az udvar másik oldalára, a bokrok mögé szeretnék kormányozni.

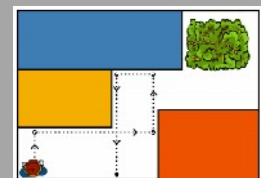
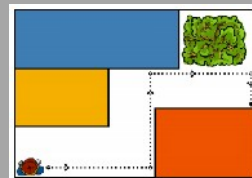
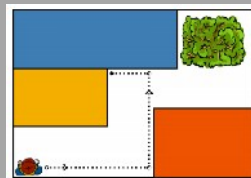
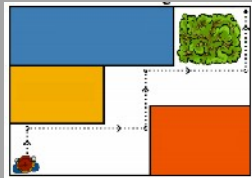


Melyik parancssorozatot kiáltják a gyerekek, hogy Bencét a bokrok mögé kormányozzák?

- A.) Előre! Jobbra! Előre! Balra! Előre! Jobbra! Előre! Balra! Előre!
B.) Jobbra! Előre! Balra! Előre! Balra! Előre!
C.) Jobbra! Előre! Balra! Előre! Jobbra! Előre! Jobbra! Előre!
D.) Előre! Jobbra! Előre! Balra! Előre! Balra! Előre! Balra! Előre!

„A” válasz a helyes:

Az ábrák mutatják az egyes megoldások eredményét:



Ez informatika!

A feladat a robotok vezérlésével foglalkozik. Azon parancsokat, melyeket egy robot követni tud, az robot programozási nyelvénél nevezzük. A parancsok sorozata pedig a robot programja. Sok különböző fajta robot létezik: vannak például kerekeken guruló, propellerrel repülő, egy lábon ugrándozó, több lábon futó, vízbe merülő, világűrben száguldozó robotok is. Néháynak vannak kezei és markológépei. Némelyek kamerával látnak, mikrofonnal hallanak, tapogatóval éreznek. Minél sokoldalúbbak egy robottípusnak az érzékelői és Aktorik-jai, annál sokoldalúbb a programnyelve.

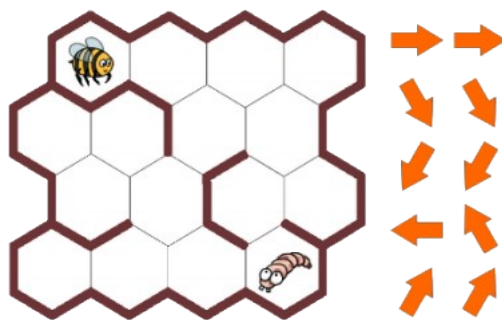


benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Méhkaptár (2013-SK-06)

A méhek  feladata, hogy etessék lárváikat . A kaptárban a méhek egyik lépsejtről  a másikkra közlekednek. Néhány helyen falak  zárják el az utat.

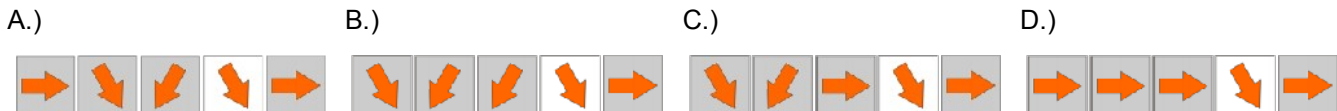
A nyilak azt mutatják, hogy a méhecske melyik irányba halad tovább a következő lépsejt felé.



A méhecskének 5 lépésre van szüksége, hogy megérkezzen a lárvához. A negyedik lépést már megadtuk.



Az alábbi megoldások közül melyik úton érhet el az éhes lárvához?



„A” válasz a helyes:

3 lehetséges megoldással érheti el a méhecske 5 lépéssel a lárvát, de csak az egyik lehetőségénél vezet a negyedik lépése jobbra-lefelé.

Ez informatika!

A nyilak sorrendje egy egyszerű program, aminek a segítségével a méhecske meg tudja találni a célját. A számítógépes programok is utasításokból állnak, melyek a számítógépnek pontosan előírják, mit kell tennie. A programozás során pontosan el kell tudnunk képzelni, hogy hatnak az utasításaink, és hogy az utasítások milyen sorrendben vezetnek a kívánt eredményhez.

	benjamin	nehéz	közepes	könnyű
	kadét	nehéz	közepes	könnyű
	junior	nehéz	közepes	könnyű
	senior	nehéz	közepes	könnyű

Pöttyök és parancsok (2013-SK-09)

Három parancs („draw-1”, „draw-2a”, „draw-2b”) segítségével a következő mintákat lehet létrehozni:



draw-1



draw-2a



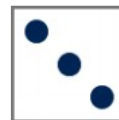
draw-2b

A „turn90” parancs az addig létrehozott mintát forgatja el 90 fokkal az óramutató járásával megegyező irányba.

Azzal, hogy több parancsot adunk egymás után, sokféle mintát hozhatunk létre. A „draw-2b, turn90” parancssor például a következő mintát eredményezi:



A „draw-1, draw-2a, turn90” parancssor pedig ezt a mintát hozza létre:

**Melyik parancssor eredményezi a következő mintát?**

- A.) „draw-2b, turn90, draw-2a, draw-1”
- B.) „draw-2b, draw-2a, turn90, draw-2a”
- C.) „draw-2a, draw-2b, turn90, draw-2a”
- D.) „draw-2a, turn90, draw-2a, draw-2b”

„D” válasz a helyes:



„A” parancssor eredménye:



„B” parancssor eredménye:



„C” parancssor eredménye:

Ez informatika!

Ennek a mintanyelvnek a kifejezési lehetőségei elég korlátozottak. Nincsenek ismétlődő ciklusok, nincsenek feltételes elágazások, nincsenek paraméterek, nincsenek [Datenobjekte].

Az biztos, hogy ez a mintanyelv nem „univerzális”. Csúpn a 4 parancsot (draw-1, draw-2a, draw-2b, turn90) lehet tetszés szerint egymás után rakni, és ezzel a háromszor három pontból álló mátrixot megváltoztatni.

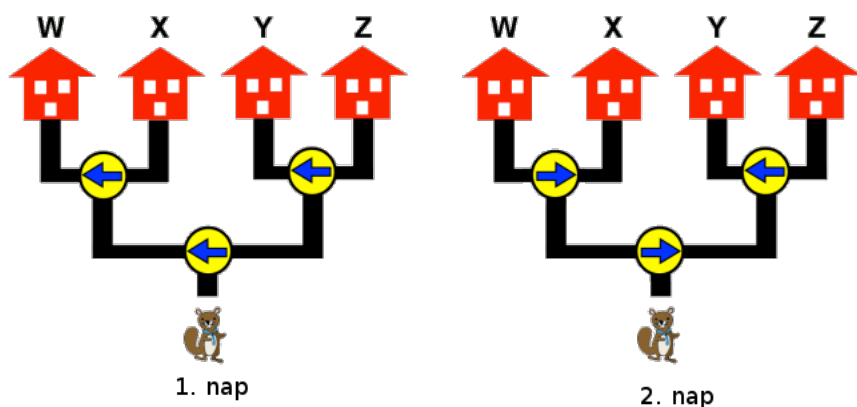
A programozáshoz szükséges gondolkodást azonban lehet vele fejleszteni: ki tud előállítani egy adott mintát a legrövidebb programmal.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Baráti látogatás (2013-TW-04)

Hód úrnak négy barátja van, akik más-más házban laknak. Mindennap meglátogatja egyiküket. Ha Hód úr az odafelé úton elágazáshoz ér, a nyíl irányába halad tovább, de egyúttal át is forgatja azt az ellenkező irányba. Hód úr az első napon W-t látogatja meg. Mivel ezen az úton két nyilat is az ellenkező irányba forgat, a második napon Y-t látogatja meg. És így tovább.



Kit látogat meg Hód úr a 30. napon?

- A.) W-t
- B.) X-t
- C.) Y-t
- D.) Z-t

„C” válasz a helyes: A 30. napon Y-t látogatja meg Hód úr.

Hód úr egy elágazásnál mindig balra fordul, ha páratlan az a szám, ahányadszor eléri az adott elágazást. Mindig jobbra fordul, ha ez a szám páros. A 30. napon 30. alkalommal (páros) érinti az első elágazást, így jobbra fordul. A következő elágazást 15. alkalommal (páratlan) érinti, így balra fordul.

Egy másik módszer: az 5. napon az összes nyíl ugyanabba az irányba mutat, mint az első napon. A nyilak helyzete 4 naponként ismétlődik. A 30. napon tehát a 2. nap fog megismétlődni ($30:4=7, 2$ a maradék).

Ez informatika!

A feladatot egy top-down-analízis segítségével lehet megoldani: ez elárulja, hogy melyik kereszteződésben merre kell tovább menni.

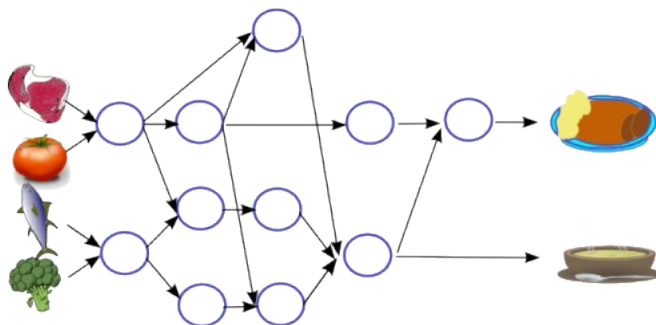
Emellett a nyilak periodikusan ismétlődő helyzetéből következtethetünk a megoldásnál, ami az informatikában egy gyakran alkalmazott technika. Ebben az esetben modulo-számítások segítségével juthatunk el a megfejtésig.



benjamin	nehéz	közepes	könnyű
kadét	nehéz	közepes	könnyű
junior	nehéz	közepes	könnyű
senior	nehéz	közepes	könnyű

Hatékonyan a konyhában (2013-TW-08)

Anna és Beni éhesen jönnek haza, és a lehető leggyorsabban szeretnének megvacsorázni. A konyhában brokkolit, halat, paradicsomot és húst találnak. Ebből két fogást szeretnének készíteni. A vacsorakészítés több lépésben történik. A legtöbb lépést csak azután tudja Anna és Beni elkezdni, miután más lépéseket már elvégzett. A képen a lépéseket folyamatábrába rendezve láthatod, a lépések sorrendjét nyilak jelölik.



Anna és Beni tűzhelyén három főzőlap van, szóval egyszerre legfeljebb három lépést tudnak elvégezni. Minden lépéshez 5 percre van szükségük.

Legkevesebb hány percre van szükségük a két fogás elkészítéséhez?

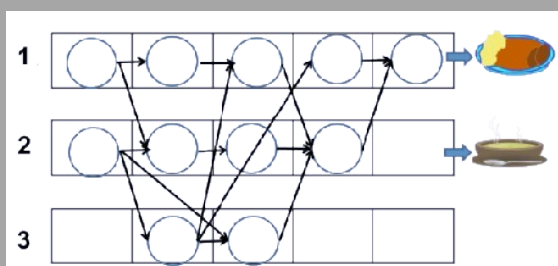
- A.) 15 perc
- B.) 20 perc
- C.) 25 perc
- D.) 30 perc

„C” válasz a helyes:

A képen láthatod, hogy lehet elosztani a lépéseket a három főzőlapon, a lehető leggyorsabb vacsorakészítés érdekében. Így az egyes főzőlapot 5 lépésre használjuk. Így adódik össze a 25 perc, mint minimális főzési idő.

Ez informatika!

Ha egy számítógép által végrehajtandó programot több processzorra akarunk osztani, akkor a programot megfelelő részekre kell bontanunk. A processzorok között történő felosztásnak úgy kell történnie, hogy a program részleteknek a lehető legkevesebb ideig kelljen más programrészletek részeredményeire várnia. Az informatika mindig egyre jobb és jobb algoritmusokon dolgozik a munkafázisok hatékony beosztásának érdekében.



	Könnyű	Közepes	Nehéz
Benjamin	2008-DE-02 2013-CA-06 2013-HU-01 2013-SK-01 2013-SK-04 2013-SK-06	2010-LT-05 2011-LT-14 2012-FR-06 2013-JP-02 2013-RU-05 2013-JP-07	2013-AT-11 2013-BE-09 2013-CH-02 2013-DE-05 2013-SI-05 2013-SI-09
Kadét	2012-FR-06 2013-CA-06 2013-JP-02 2013-RU-05 2013-SI-01 2013-SI-04	2010-LT-05 2013-SI-09 2013-AT-04 2013-JP-04 2013-JP-07 2013-SI-05	2012-DE-02 2013-CH-01 2013-CH-03 2013-FR-02 2013-HU-04 2013-NL-06
Junior	2012-DE-02 2013-AT-04 2013-HU-07 2013-JP-04 2013-JP-07 2013-SI-04	2013-CH-03 2013-CH-04 2013-DE-02 2013-FR-02 2013-SI-09 2013-SK-09	2012-NL-02 2013-AT-08 2013-AT-09 2013-CH-01 2013-NL-02 2013-TW-04
Senior	2012-DE-02 2013-CH-02 2013-CH-03 2013-CH-04 2013-SI-09 2013-JP-07	2013-AT-08 2013-AT-09 2013-DE-02 2013-NL-02 2013-TW-04 2013-TW-04	2011-DE-14 2011-NL-01 2012-NL-02 2013-BE-07 2013-IT-07 2013-LV-02

Támogatóink:

