



Međunarodno takmičenje iz informatičke i računarske pismenosti

„Dabar“

Zadaci, tačni odgovori i informatičke pozadine zadatka sa
školskog takmičenja 2015/2016.god.

Kategorije:

- Dabarčić (5. i 6. razred osnovne škole)
- Mladi dabar (7. i 8. razred osnovne škole)
- Dabar (1 i 2. razred srednje škole)
- Stariji dabar (3 i 4. razred srednje škole)

Kruševac, Novembar 2015.god.

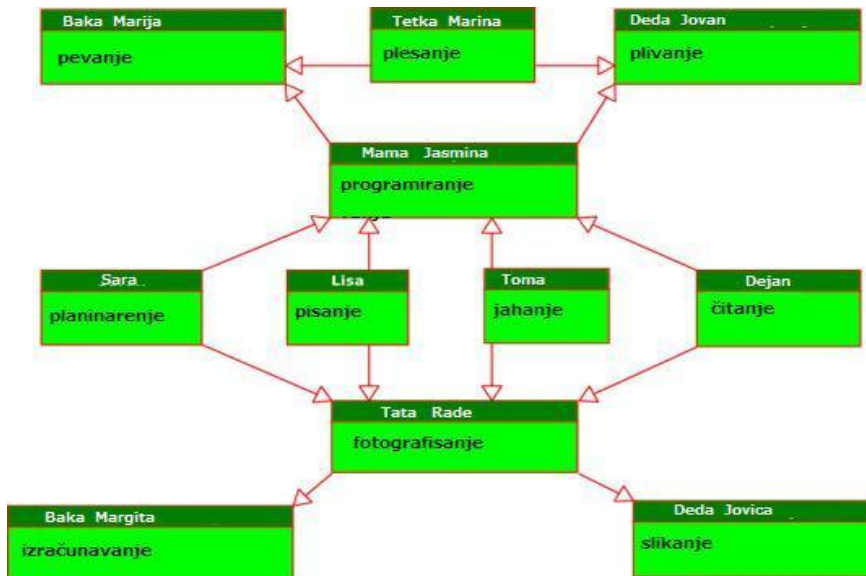
Pitanja

Moćna porodica dabrova	3
Telefonski imenik	4
Auto robot	5
Predak	6
Upravljanje kranom	7
Redosled olovaka	8
Elektronska pošta	9
Zalivanje bašte	10
Visina životinja	11
Piramida brojeva	12
Dabrova kućica	13
Robot i kocke	14
Sakupljanje grana	15
Sekvence	16
Mobilni	17
Sustići	18
Banka	19
Karte sa brojevima	20
Kengur koji skače	21
Uređivanje redosleda vagona	22
Konjske trke	23
Pirati	24
Izaberi kamenje	25

Moćna porodica dabrova

U porodici dabrova svi članovi imaju određene osobine. Čerka nasleđuje sve osobine od majke; sin nasleđuje sve osobine od svog oca. Pored nasleđenih osobina, svaki član porodice ima jednu dodatnu osobinu. Dijagram prikazuje odnose između dabrova kao i dodatnu osobinu za svakog dabra.

Na slici možete videti da je majka Jasmina nasledila osobinu da peva od bake Marije, a pored toga ima osobinu da programira. Liza nasleđuje od majke obe ove osobine, ali ima i osobinu pisanja. Dakle, ona ima osobine da pisanja, programiranja i pevanja.



Pitanje

Koja izjava je tačna?

1. Sara ima sledeće osobine: planinarenje, programiranje i plivanje.
2. Toma nasleđuje od bake Margite osobinu izračunavanja.
3. Tetka Marina ima osobine: plesanje i plivanje.
4. Tomine osobine su jahanje, slikanje i fotografisanje

Odgovor: Izjava 4 je tačna.

Informatička pozadina: Objektno orijentisano programiranje se zasniva na različitim tehnikama kao što su nasleđivanje, modularnost, polimorfizam i enkapsulacija.

Nasleđivanje predstavlja mogućnost hijerarhijske organizacije klasa. Kada jedna klasa nasledi drugu, ona zadržava kompletan sadržaj klase koju nasleđuje i taj sadržaj može redefinisati ili proširiti. Postoji nekoliko vrsta nasleđivanja.

U našem zadatku sposobnosti su kao delovi softvera koji se nasleđuju i obogaćuju dodatnim sposobnostima.

Telefonski imenik

Dabar Milan traži broj telefona njegove prijateljice iz Dabrograda na veb stranici na kojoj se nalazi veoma veliki broj imena i brojeva telefona.

Dabar Milan nije siguran kako se tačno piše ime njegove prijateljice tako da, prilikom pretrage veb stranice, koristi specijalne karaktere (?, & i %):

karakter ? koristi se kada tačno jedno slovo nije poznato

karakter & koristi se kada su tačno dva uzastopna slova nisu poznata

karakter % koristi se kada ostatak imena nije poznat

Na primer, Te% daje rezultate pretrage Tereza, Teodor, Teofil itd.

Milan je u pretrazi napisao sledeće: **S?rah B&cht%**

Pitanje

Koje ime Milan traži?

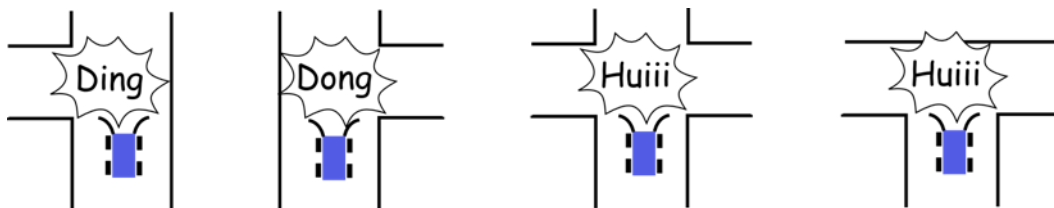
Sarah Birchman
Sarah Bilchdrain
Sarah Birchtree
Sarah Bechtree

Odgovor: Sarah Birchtree

Informatička pozadina: Ovo je klasičan primer problema u pronalaženju određenog dela teksta kada možemo koristiti tzv. džoker-karakter.

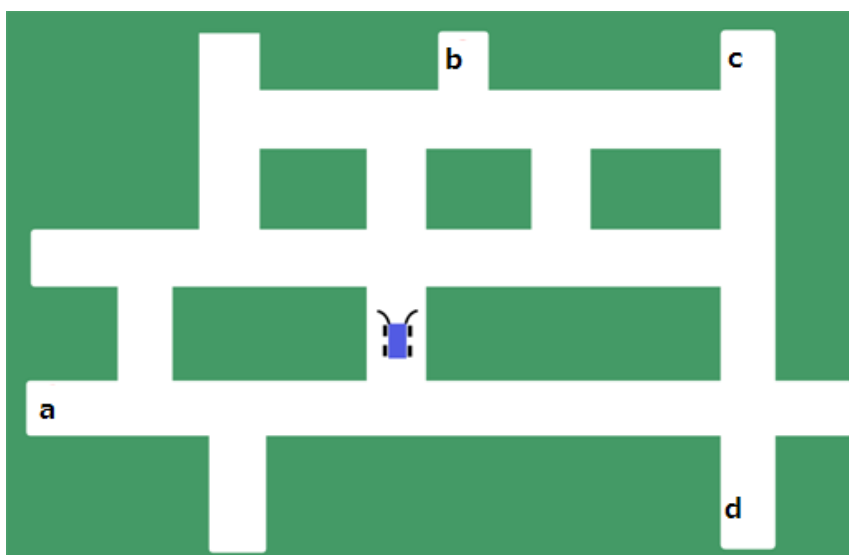
Auto robot

Fabrika automobila Dabrocedes je napravila robot-automobil. Robot-automobil ima i senzore koji detektuju raskrnicu. Kada naiđe na raskrnicu, robot-automobil proizvodi zvukove. Robot-automobil može da ide pravo, levo i desno.



Dabrica Ceca je ušla u robot-automobil da se provoza i robot-automobil je proizveo sledeći zvuk:

Huiii Ding Huiii Dong



Pitanje

Na koju je poziciju robot-automobil odvezao dabricu Cecu?

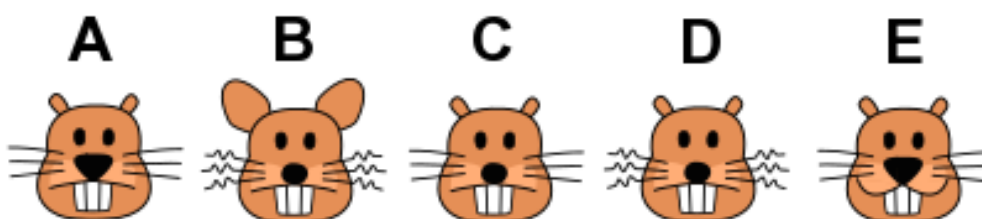
- a
- b
- c
- d

Odgovor: b

Informatička pozadina: Razvoj sistema za automatizovanu kontrolu vozila je važno polje u informatici.

Predak

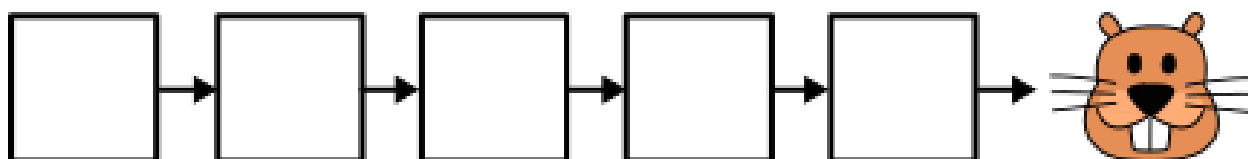
Dabar Bibi želi da uradi svoje porodično stablo. Sakupio je fotografije svojih predaka, pet generacija unazad, kao na slici ispod.



U Bibinoj familiji dete nasleđuje lice roditelja osim jednog detalja.

Pitanje:

Na slici ispod Bibijeva fotografija je poslednja. Po kom redosledu treba poređati slike Bibijevih predaka?



- D -> B -> E -> C -> A
- B -> D -> A -> C -> E
- D -> B -> C -> E -> A
- B -> D -> C -> A -> E

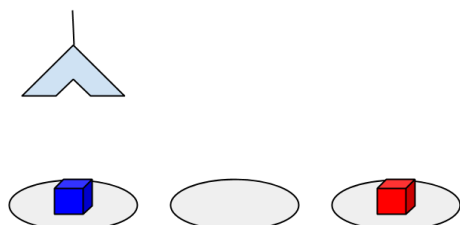
Odgovor: B -> D -> C -> A -> E

Informatička pozadina: Ukoliko želite da pošaljete ove slike dabrova preko mreže, možete ih poslati sve. Međutim, možete da smanjite kapacitet prenosa ako pošaljete prvu sliku i svaku razliku između susednih slika. To je primer "kompresije podataka". Kompresija podataka je važna jer pomaže da se smanji upotreba resursa, kao što su skladišta ili prenosni kapacitet.

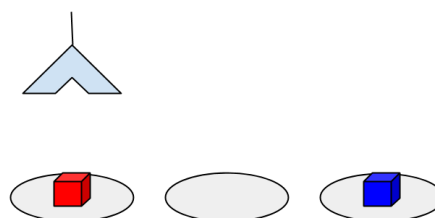
Upravljanje kranom

Dabrica Sandra radi na dizalici (kranu). Prilikom rada na dizalici postoji šest komandi "Dole", "Gore", "Desno", "Levo", "Uhvati", i "Pusti".

Početak



Kraj



Pitanje

Koji redosled komandi Sandra treba izabrati da bi stanje sa početka dovela u stanje na kraju (slika iznad)?

Dole-Uhvati-Gore-Desno-Desno-Dole-Pusti-Uhvati-Gore-Levo-Levo-Dole-Pusti-Gore

Desno-Desno-Dole-Uhvati-Gore-Levo-Levo-Dole-Pusti-Uhvati-Gore-Desno-Desno-Dole-Pusti-Gore

Dole-Uhvati-Gore-Desno-Dole-Pusti-Gore-Desno-Dole-Uhvati-Gore-Levo-Levo-Dole-Pusti-Gore-Desno-Dole-Uhvati-Gore-Desno-Dole-Pusti-Gore-Levo-Levo

Dole-Uhvati-Gore-Desno-Dole-Pusti-Gore-Desno-Dole-Uhvati-Gore-Levo-Dole-Pusti-Gore

Odgovor: Dole-Uhvati-Gore-Desno-Dole-Pusti-Gore-Desno-Dole-Uhvati-Gore-Levo-Levo-Dole-Pusti-Gore-Desno-Dole-Uhvati-Gore-Desno-Dole-Pusti-Gore-Levo-Levo

Informatička pozadina: Ovaj zadatak ima dodirnih tačaka sa algoritmima, programiranjem i privremenim smeštanjem podataka.

Algoritam je opis za rešavanje nekog problema. U ovom zadatku potrebno je rešiti kako promeniti mesta plavom i crvenom objektu. Za čoveka je laka razmena dva objekta istovremeno jer ima dve ruke. Međutim, računari mogu obraditi jedan po jedan korak. U ovom slučaju plavi ili crveni objekat će biti privremeno postavljen na sredini (u informatici to se naziva **privremeno smeštanje podataka**). Nakon toga, drugi predmet se premešta na prvobitnu poziciju prvog predmeta. I na kraju, predmet na sredini će biti pomeren. Da bi dizalica radila, moraju se zadati konkretne komande (u informatici se to naziva **programiranje**).

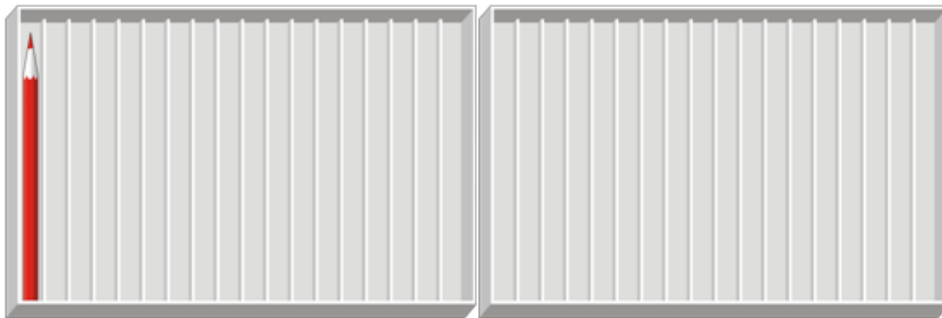
Redosled olovaka

Mali dabar ima punu kutiju korišćenih olovaka. Njemu je dosadno da crta i želi da se igra olovkama.



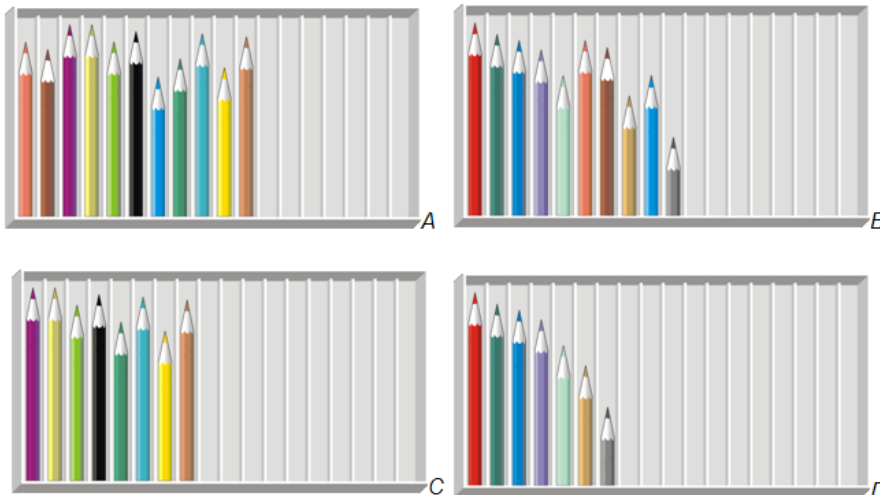
Počinje igru prema sledećim pravilima:

- Olovke uzima iz kutije jednu po jednu sa leve na desnu stranu.
- Olovke raspoređuje u maminu i tatinu kutiju za olovke, takođe sa leve na desnu stranu.
- Prvu olovku stavlja u maminu kutiju za olovke.
- Svaku sledeću olovku poredi sa poslednjom koju je smestio u maminu kutiju za olovke – ako nije duža od te poslednje, on je, takođe, smešta u maminu kutiju za olovke, ukoliko je duža, smešta je u tatinu kutiju za olovke.



Pitanje

Koji raspored olovaka će biti u tatinoj kutiji za olovke, kada mali dabar spakuje poslednju olovku?



Odgovor: A

Informatička pozadina: Pravila ove igre su primer algoritma koji treba da se sledi da da bismo dobili odgovor na pitanje. Prilikom pisanja kompjuterskih programa uobičajeno je da se slede određeni algoritmi. Osim toga, ovo je, iako deluje neuobičajeno, takođe algoritam za sortiranje i oni se koriste u informatici vrlo često.

Elektronska pošta

Dabar Bojan je u potrazi za novom kućom. Pretražio je dosta internet stranica sa raznim ponudama i našao je **savršenu kuću po veoma povoljnoj ceni**.

On je poslao elektronsku poštu (e-mail) dabru Vučku, koji prodaje stan preko internet stranice i dobio, veoma brzo, odgovor:

Zdravo Bojane,

hvala na interesovanju za moj stan. Iako nisam u gradu, mogu ti brzom poštom poslati ključ od stana tako da možeš pogledati stan. Ali moram biti siguran da me nećeš prevariti i tražim unapred depozit od 5.000 dinara.

Da bi i ti bio siguran da ja neću tebe prevariti – poslaću ti kopiju moje lične karte.

Srdačan pozdrav, Vučko.

Pitanje

Bojan nije siguran šta da radi i traži vašu pomoć. Koji će biti vaš najbolji savet?

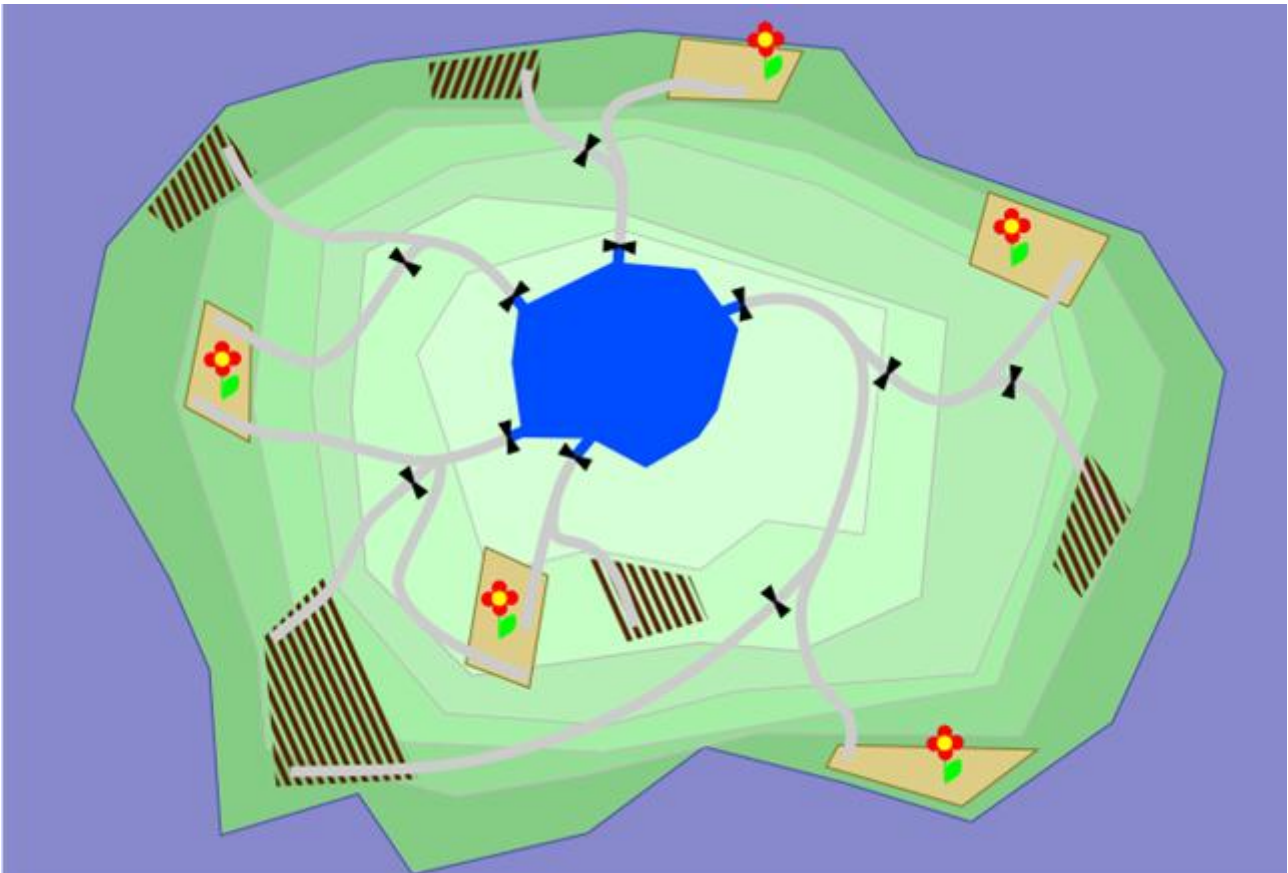
1. Ovo je savršeno. Uplati depozit i ako želiš stan, možeš zadržati ključ odmah.
2. Uplati unapred depozit, idi i pogledaj stan, pa odluči kasnije.
3. Nemoj da uplaćuješ depozit, postoji velika šansa da je ovo prevara.
4. Vučko će poslati kopiju lične karte. Ako ti se ne sviđa stan, a on ti ne vrati depozit, prijavićeš ga policiji i dobićeš novac nazad.

Odgovor: Savet broj 3

Informatička pozadina: Internet je veoma pogodan za prikriivanje identiteta i obezbeđuje anonimnost. Kriminalci koriste ovaj mehanizam i raznim prevarama dobijaju novac od lakovernih ljudi. U takvim prevarama (na društvenim mrežama, internet stranicama ili u elektronskoj pošti) vrlo često postoje pravopisne greške i pominju se velike novčane vrednosti.

Zalivanje bašte

Dabrovi treba da zaliju svoje bašte. Danas će zalivati samo bašte u kojima je cveće. Odvijanjem ventila (koji je prikazan crnom bojom) moguće je pustiti vodu kroz cevi. Ako je odvijen ventil, voda će doći do bašte (iz jezera koje je na vrhu brda).



Pitanje

Koliko najmanje ventila treba otvoriti da bi se zalili sve bašte sa cvećem?

- 4 ventila
- 6 ventila
- 5 ventila
- 7 ventila

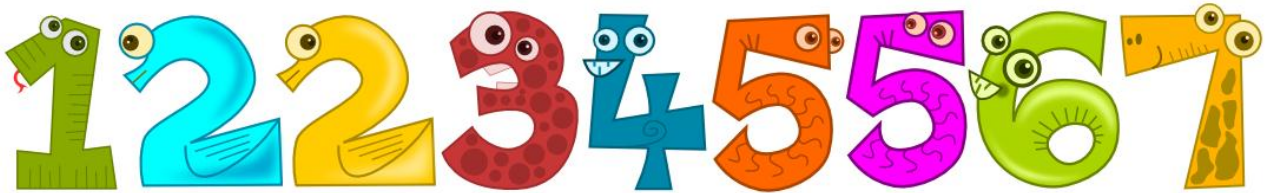
Odgovor: 4 ventila

Informatička pozadina: U ovom zadatku se traži najmanji broj ventila koje treba otvoriti da bi se zalile sve bašte sa cvećem. Prilikom programiranja se vodi računa o minimizaciji vremena izvršavanja programa, korišćenja memorije itd.

Visina životinja

Životinje iz šume su merile svoju visinu. Ove vrednosti su zabeležene u decimetrima: 1, 2, 2, 3, 4, 5, 5, 6, 7.

Znamo da nijedna devojčica nije viša od nekog od dečaka. Ljilja je visoka koliko i Milan, a



Sofija je visoka koliko i Marija.

Pitanje

Koliko devojčica ima u šumskom razredu?

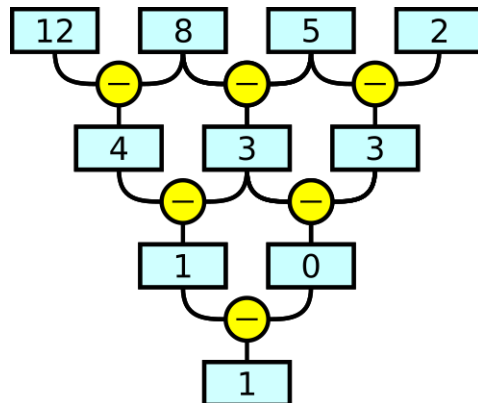
- 2
- 3
- 6
- 7

Odgovor: 6

Informatička pozadina: pri radu sa podacima neka organizacija je neophodna. Ovaj zadatak zahteva da shvate kako su podaci poređani i kako se primenjuju zahtevana pravila. Takođe je potrebno da koristite neku logiku da rešite ovaj zadatak. Zapamtite da nije matematika neophodna za rešavanje ovog zadatka. Možete ga rešiti čak i da su bila napisana slova umesto brojeva: A, B, B, C, D, E, E, F, G.

Piramida brojeva

Mašina ispod uzima 4 broja u prvom redu kao ulaz. U svakom redu računa razlike. Evo primera:



Pitanje

Koji od navedenih ulaznih brojeva će rezultirati nulom u zadnjem redu?

13 9 7 6

16 9 4 1

13 8 4 2

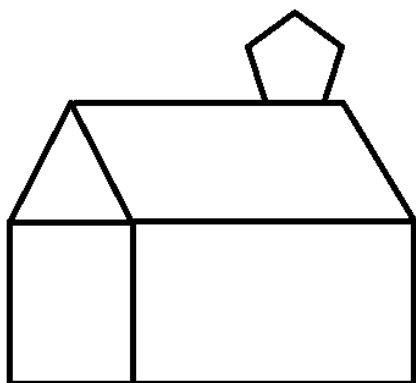
5 5 5 1

Odgovor: 16 9 4 1

Informatička pozadina: ova mašina sprovodi jednostavne proračune. Slični proračuni se dešavaju i u kompjuterskom čipu, gde se ulazne informacije obrađuju u par koraka da bi se dobile izlazne (rezultati). Ovaj proračun može, takođe, da se progra,mira veoma jednostavno, pomoću tabele.

Dabrova kućica

Dabar gradi kuću kao što je prikazano na slici.



Pitanje

Koliko je različitih geometrijskih oblika i štapića potrebno da bi sagradio kuću?

5 oblika, 17 štapića

6 oblika, 15 štapića

6 oblika, 17 štapića

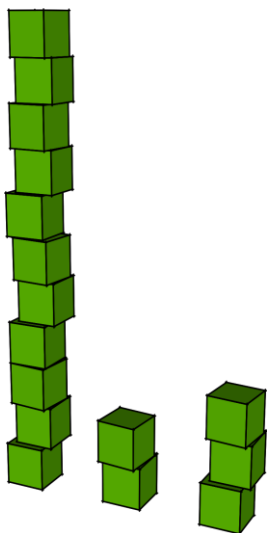
5 oblika, 15 štapića

Odgovor: 5 oblika, 15 štapića

Informatička pozadina: to je informatika, jer se za rešenje problema koriste aranžmani.
(Strukture, obrasci i aranžmani)

Robot i kocke

Robot obavlja posao vršeći određenu radnju. Radnja se sastoji od dva koraka. Robot treba da uzme kocku sa jedne kule i da je premesti na neku od susednih kula. Robot treba da izjednači broj kocki u kulama što je više moguće.



Pitanje

Koliko puta treba da ponovi radnju da bi izvršio zadatak?

- 4
- 5
- 6
- 7

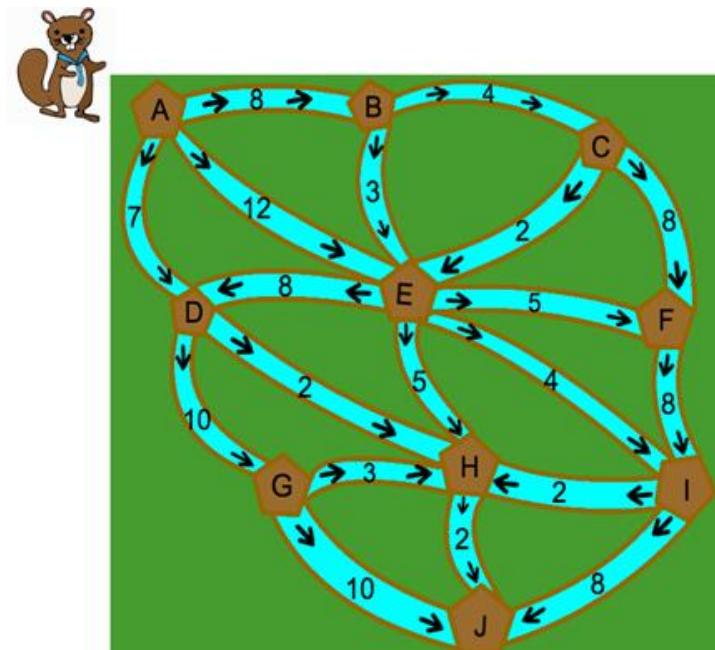
Odgovor: 5

Informatička pozadina: jednostavan algoritam i komparacija

Sakupljanje grana

Izgradnja brane je najvažniji godišnji zadatak za dabrove. Oni prikupljaju mnogo grana iz šume za gradnju. Da bi prevezli te teške grane, pametni dabrovi koriste rečne sisteme kanala na osnovu sledećih pravila:

1. Dabrovi seku grane i bacaju grane u kanale.
2. Dabrovi kroz kanale plove brodom nizvodno i kupe grane.
3. Dabrovi samo jednom idu od mesta A do mesta J i traže put sa najviše grana.



Pitanje

Ako je situacija kao na slici, koliko će najviše grana sakupiti dabrovi od mesta A do mesta J?

36

39

40

42

Odgovor: 42 grane

Informatička pozadina: pretraga u dubinu (na engleskom Depth-first search - DFS) je algoritam za pretragu struktura podataka (stabla i grafova). Početak algoritma je u korenu stabla (kod grafa se neki čvor odredi za koren), a zatim se pretražuje duž svih grana koliko god je to moguće pre povratka u koren.

Sekvence

Dabar Branko želi da sazna nešto o programiranju. Dabar Bojan mu je pokazao deo svog programskog koda:

```
x ← 3 · 4 + 2  
x ← 0  
y ← 2 · x - 3  
x ← x + 3
```

Pitanje

Koja je vrednost x i y nakon izvršenja ovog programskog koda?

14 i 17
10 i 13
-8 i -5
3 i -3

Odgovor: 3 i -3

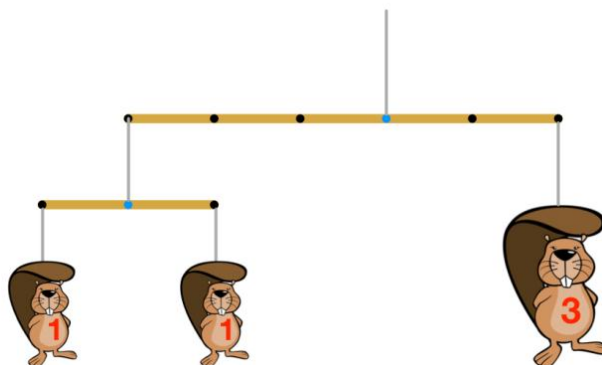
Informatička pozadina: u programiranju varijable su dinamične, tj. menjaju vrednost tokom vremena. U našem zadatku promenljiva X menja vrednost tri puta (14, 0 i 3). Pseudokod je programski kod napisan običnim jezikom koji se koristi u udžbenicima programiranja da bi se objasnio određeni algoritam bez korišćenja konkretnog programskog jezika.

Mobilni

Ne, u ovom zadatku se ne radi o mobilnim telefonima!

Reč je o mobilnim (pokretnim) igračkama koje su visile sa plafona vaše spavaće sobe kada ste bili dete.

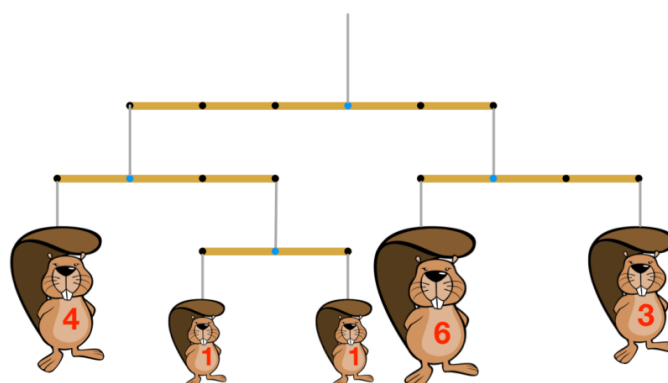
Slika ispod pokazuje jednostavnu mobilnu igračku:



Ova jednostavna mobilna igračka se može predstaviti pomoću brojeva i zagrada:

$(-3 (-1 1) (1 1)) (2 3)$

A sada, evo nove mobilne igračke:



Pitanje

Kako ćemo predstaviti brojevima i zagradama ovu igračku?

$(-3 (-1 4) (2 (-1 1) (1 1))) (2 (-1 6) (2 3))$

$(4 (1 1)) (6 3)$

$(((-1 4) -3 ((-1 1) 2 (1 1)))((-1 6) 2 (2 3)))$

$(-3 (-1 4) (2 (-1 1) (1 1)) (2 (-1 6) (2 3))$

Odgovor: $(-3 (-1 4) (2 (-1 1) (1 1))) (2 (-1 6) (2 3))$

Informatička pozadina: struktura ove mobilne igračke je zanimljiva: ako otkačimo jedan štap (osim onog najvišeg), imaćemo jednu manju mobilnu igračku i tako opet... U informatici je ovakva pojava poznata kao rekurzija. Rekurzija u informatici označava postupak ili funkciju koji u svojoj definiciji koriste sami sebe. Drugim rečima, ukoliko neki postupak zahteva da delovi problema koje je razdvojio od drugih bivaju nezavisno podvrgnuti istom tom postupku, taj postupak je rekurzivan.

Sustići

Rumenka Dabar ima gomilu od 10 stabala. Zlata Dabar ima samo jedno stablo.



Rumenka Dabar i Zlata Dabar počinju istovremeno da žvaću koru drveća u šumi. Svako drvo koje zagrizu dodaju svojoj gomili stabala.

Rumenka Dabar žvaće jedno drvo na sat.

Zlata Dabar žvaće progresivno: u prvom satu žvaće jedno drvo, u drugom satu žvaće dva, u trećem tri itd.

Pitanje

Za koliko najmanje sati će Zlata imati bar isto drveća kao Rumenka?

- 4
- 5
- 6
- 7

Odgovor: 5

Informatička pozadina: pitanje ima veze sa asimptotskom analizom algoritama. Mogli bismo reći da Rumenka žvaće drveće sa linearnom stopom rasta, što se zapisuje kao $O(n)$. Dakle, posle n sati, imaće ukupan broj stabala koji je proporcionalan n . Za Zlatu možemo reći da žvaće drveće sa kvadratnom progresijom, što zapisujemo $O(n^2)$, što znači da će posle n sati imati ukupan broj stabala proporcionalan n^2 .

Banka

U zemlji Dabroviji, imaju valutu sa čudnim novčićima. Vrednosti novčića su:

- 1 cent
- 7 centi
- 12 centi
- 22 centa

Ti radiš u banci, dolazi ti mušterija koja želi da podigne 39 centi.

Pitanje

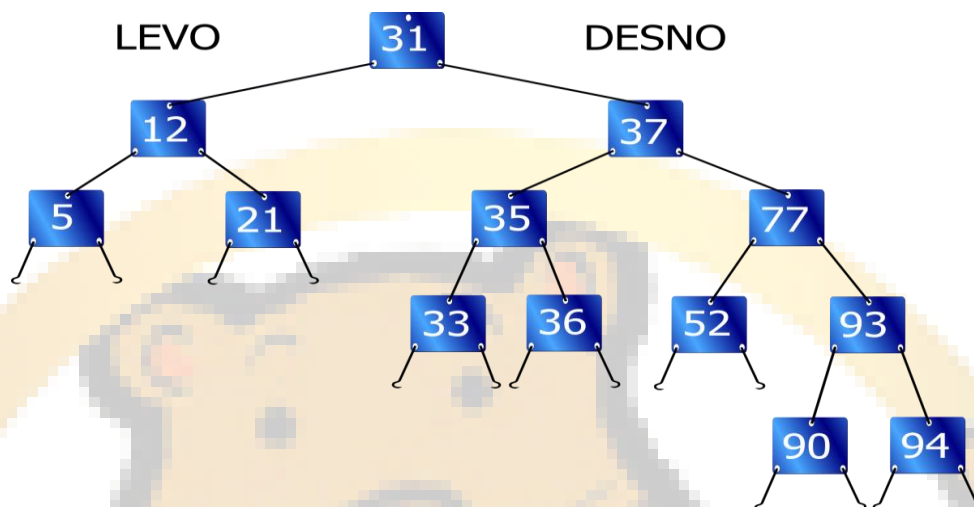
Koliko najmanje ukupno novčića ćeš dati mušteriji?

- 5
- 6
- 7
- 4

Odgovor: 5

Informatička pozadina: grabljivi algoritam se ne može direktno primeniti u ovom zadatku. Mnogi problemi u računarstvu mogu da se rešavaju koristeći grabljivi pristup, ali sa pažljivijom analizom, grabljivi pristup daje optimalno rešenje. Drugi pristup u takvim situacijama je dinamičko programiranje.

Karte sa brojevima



Napravljena je struktura od karata sa brojevima kao na slici. Svaka karta sa brojem napravljena je na sledeći način:

- Ima otvor na vrhu i dve žice na dnu koje se mogu povezati sa otvorom na vrhu ostalih karata. Ove karte se zovu karte sledbenici.
- Ceo broj N je napisan na svakoj karti.

Ako se karta sa brojem poveže na levu žicu, broj na njoj i brojevi na svim njenim kartama sledbenicima mora biti manji od N.

Ako se karta sa brojem poveže na desnu žicu, broj na njoj i brojevi na svim njenim kartama sledbenicima mora biti veći od N.

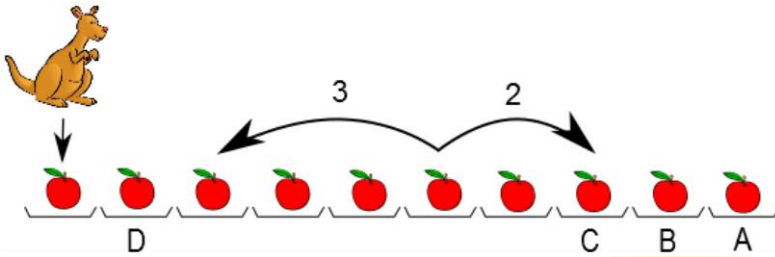
Pitanje:

Neke karte sa brojevima imaju slobodne žice. Koliko karata se može dodati da se proširi struktura?

Odgovor: 11

Informatička pozadina: struktura karata sa brojevima naziva se binarno stablo za pretraživanje (binary search tree). Ona liči pomalo na stvarno stablo okrenuto naopako. Naziva se binarno (latinska reč "bis" znači "dvaput") zato što svaka karta sa brojem ima dve žice i tako može biti povezana sa najviše dve druge karte sledbenika. Binarna stabla za pretraživanje koriste se za čuvanje svih vrsta podataka. Prednost korišćenja binarnog stabla za pretraživanje je da se mogu veoma brzo pronaći sačuvani podaci.

Kengur koji skače



Postoji 10 tanjira u jednom redu.

Kengur Toma voli da skače. Prvo, on skoči na tanjir krajnje levo. Svakim sledećim skokom on ili skoči unapred dva tanjira, ili unazad tri tanjira. (Primer dva moguća skoka je prikazan strelicama na slici.)

Toma skače samo na tanjire sa jabukom. Kada skoči na tanjir, on uzme jabuku sa njega.

Pitanje:

Ako je Toma sakupio svih 10 jabuka, koju je jabuku poslednju uzeo?

- Krajnje desna jabuka
- Druga jabuka zdesna
- Treća jabuka zdesna
- Druga jabuka sleva

Odgovor: Druga jabuka zdesna

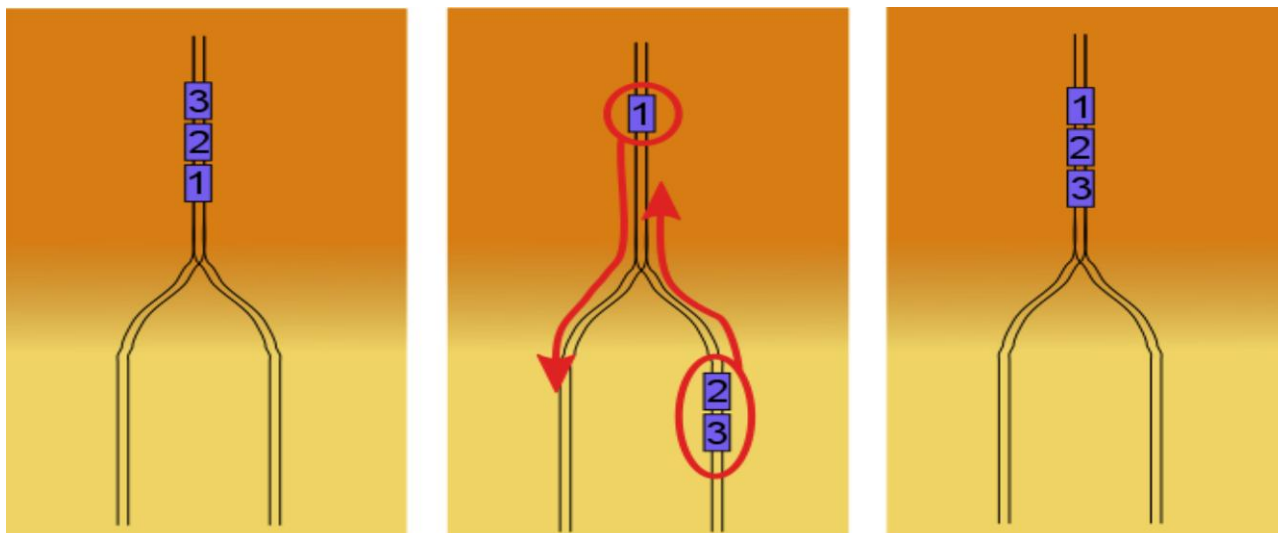
Informatička pozadina: jedan način da se reši ovaj problem je razmotriti sve moguće redoslede tanjira i tražiti onaj koji sadrži samo ispravne skokove. Svaki mogući redosled naziva se permutacija i postoji mnogo njih. Zato ovaj pristup, koji se zove algoritam grube sile (brute-force ili exhaustive search – iscrpna pretraga), zahteva mnogo vremena.

Drugi pristup je napraviti permutaciju za jedan tanjir u jednom koraku. Jednom kada shvatite da permutacija ili njen početak nisu ispravni (kao što je određivanje da drugi tanjir ne može biti nijedan osim tanjira 3), može se ukloniti poslednji tanjir(i) i nastaviti pravljenje novih permutacija. Ovo se naziva backtracking (praćenje unazad). Ako se mogu ustanoviti pravila za više permutacija u ranim fazama pretrage, mogu se brže pronaći ispravne permutacije. Ova prečica naziva se pruning (“orezivanje”).

Jedan način posmatranja ovog problema je tretirati ga kao graf. Tanjiri su čvorovi (verteksi) i dva tanjira se povezuju strelicom ako Toma može skočiti između njih. Zadatak podrazumeva pronalaženje putanje preko strelica koja posećuje svaki čvor tačno jednom. Ovo se zove Hamiltonova putanja (Hamiltonian path). U opštem slučaju veoma je teško pronaći takvu putanju. Međutim, u ovom slučaju graf je mali i ima posebna svojstva.

Za opšti problem pronalaženja Hamiltonove putanje zna se da je NP-kompletan, što znači da pripada skupu vrlo važnih problema za koje nemamo efikasno rešenje. Zanimljivo je da ako znamo da je neko pronašao rešenje za jedan od ovakvih problema, onda odmah imamo način da se efikasno reše svi ovakvi problemi.

Uređivanje redosleda vagona



Tri vagona stoje na šinama na brdu, kao što je prikazano na slici levo. Skretničar vozova želi da okrene redosled svih vagona, npr. kao što je prikazano na slici desno.

Da bi postigao ovaj cilj, skretničar može koristiti dva vozna koloseka koji idu nizbrdo i izvoditi dve vrste operacija, prikazanih na slici u sredini:

1. **Nizbrdo:** Pomeri jedan vagon sa brda nizbrdo bilo kojim od dva koloseka.
2. **Uzbrdo:** Pomeri sve vagona sa istog koloseka uzbrdo. (Npr., ako postoje dva vagona u koloseku, treba ih pomeriti zajedno.)

Operacija **Nizbrdo** je veoma laka za skretničara, vagoni idu sami, tako da on mora uraditi samo početni pomak.

Operacija **Uzbrdo** je veoma teška, zato što mora pomerati vagona uzbrdo celim putem.

Pitanje:

Koji je minimalan broj operacija **Uzbrdo** koje skretničar mora obaviti da bi postigao svoj cilj?

- 2
- 3
- 4
- 5

Odgovor: 3

Informatička pozadina: Određivanje najboljeg ili *optimalnog* rešenja problema je važna oblast proučavanja u računarstvu. Ponekad optimalna rešenja pokušavaju da minimiziraju korišćenje skupog/ređe dostupnog resursa, kao što je korišćenje što manje žice za povezivanje mreže računara. Ovaj zadatak je inspirisan voznim parkovima za razvrstavanje vagona i njihovo povezivanje u vozove. Tako da iz toga proizlazi problem optimalnog uređivanja redosleda vagona: kako urediti redosled vagona korišćenjem najmanjeg broja skupih operacija Uzbrdo. Ili obrnuto, optimalna rešenja pokušavaju da maksimalizuju, kao što je pokušaj da ima što više korisnika pokrivenih mrežom mobilnih telefona. Opšte strategije za pronalaženje optimalnih rešenja proučavaju se u naučnoj oblasti operacionih istraživanja, i mogu se napisati računarski programi da primenjuju takve strategije.

Tri lokacije za vagona mogu se zamisliti kao *magacinske memorije (stekovi)*. Stek je struktura podataka koja prati paradigmu LIFO (*Last In, First Out – Poslednji na ulazu, prvi na izlazu*). Na primer, ostava posuđa prati ovu paradigmu: može se dodati tanjir na vrh, može se skloniti tanjir sa vrha, ali se ne može dodati ili ukloniti bilo sa koje druge pozicije a da se ne izazove pad i lom posuđa. U ovom zadatku, dva koloseka nizbrdo su ograničeni magacini, budući da se operacije uklanjanja ("pop") ne mogu uraditi odvojeno jedna po jedna, budući da će se celokupni sadržaj magacina pomeriti uzbrdo u jednoj operaciji.

Konjske trke

Steva ima 25 brzih konja. Želi da odredi koja tri su najbrža. Nema štopericu. Moraće da koristi trkačku stazu, na kojoj mogu biti odjednom samo pet konja. Može se pretpostaviti da svaki konj uvek završi trku za isto vreme.

Pitanje:

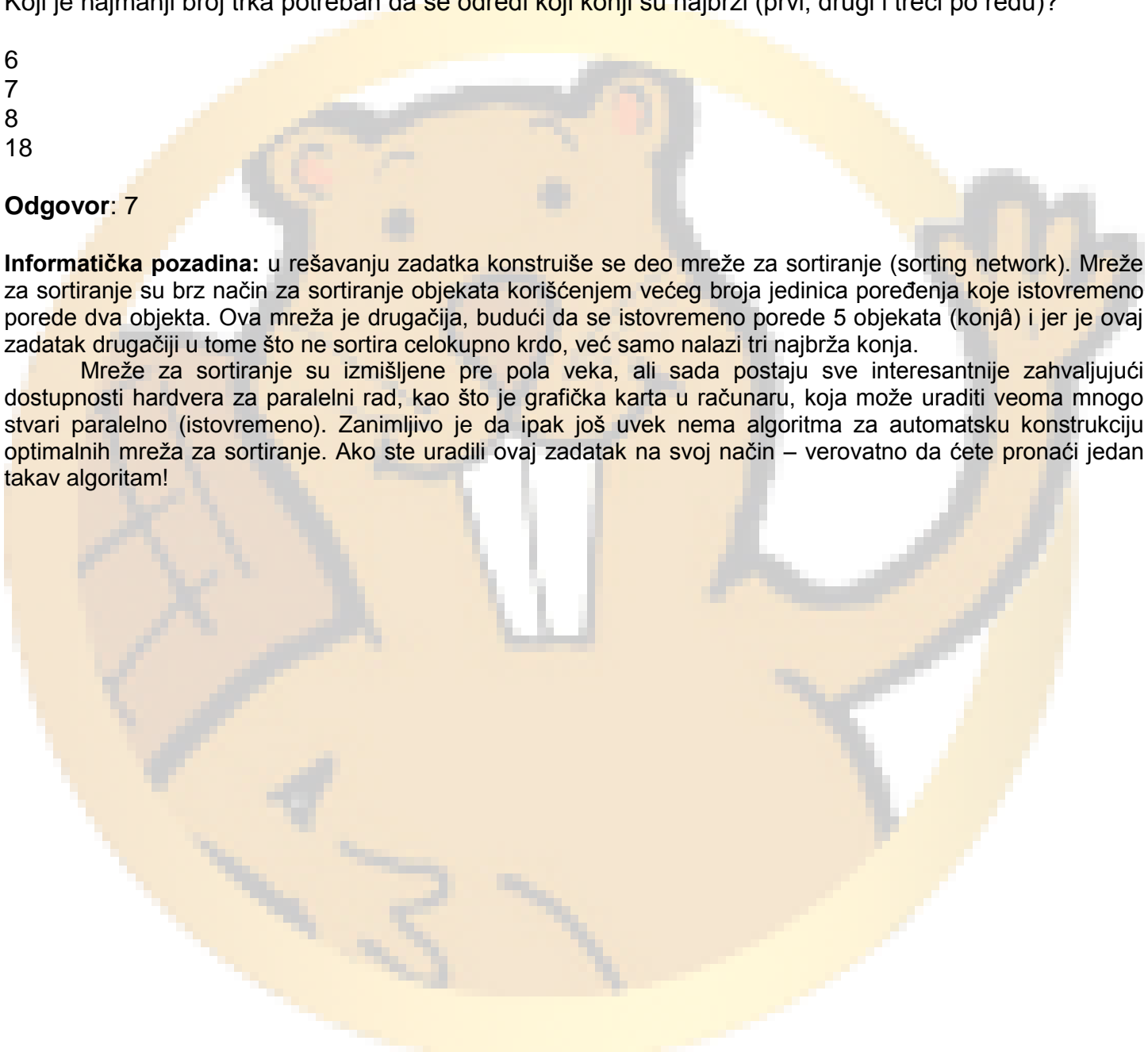
Koji je najmanji broj trka potreban da se odredi koji konji su najbrži (prvi, drugi i treći po redu)?

- 6
- 7
- 8
- 18

Odgovor: 7

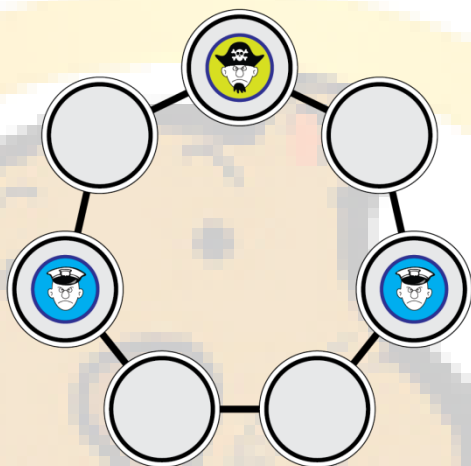
Informatička pozadina: u rešavanju zadatka konstruiše se deo mreže za sortiranje (sorting network). Mreže za sortiranje su brz način za sortiranje objekata korišćenjem većeg broja jedinica poređenja koje istovremeno porede dva objekta. Ova mreža je drugačija, budući da se istovremeno porede 5 objekata (konjâ) i jer je ovaj zadatak drugačiji u tome što ne sortira celokupno krdo, već samo nalazi tri najbrža konja.

Mreže za sortiranje su izmišljene pre pola veka, ali sada postaju sve interesantnije zahvaljujući dostupnosti hardvera za paralelni rad, kao što je grafička karta u računaru, koja može uraditi veoma mnogo stvari paralelno (istovremeno). Zanimljivo je da ipak još uvek nema algoritma za automatsku konstrukciju optimalnih mreža za sortiranje. Ako ste uradili ovaj zadatak na svoj način – verovatno da ćete pronaći jedan takav algoritam!



Pirati

Jana i Žile igraju igru Lovci na pirate (Pirate Hunters). U svakom potezu jedan od policajaca (ali ne oba) pomeraju se na susedno mesto. U sledećem potezu igra pirat koji je brži i uvek skače za dva mesta. Policajci se uvek pomeraju na nezauzeto mesto – ne mogu doći na mesto koje je zauzeo pirat ili neki kolega policajac. Igra se završava kada je pirat prisiljen da skoči na jednog od policajaca, što je prikazano na slici, osim što je trenutno policajac na potezu. Da bi pobedili, policajci moraju naterati pirata na ovu poziciju kada je pirat na potezu.



Pitanje:

Jana, koja igra pirata, veoma je vešta u izbegavanju zarobljavanja. Mada, i ti si pametan igrač. Ako pomogneš Žiletu da odigra savršenu igru, koliko poteza će on napraviti pre nego što pirat bude uhvaćen?

2

3

5

Žile ne može da pobedi

Odgovor: Žile ne može da pobedi

Informatička pozadina: programi koji igraju igre na tabli rade tako što se izračunava moguća putanja kroz graf stanja igre. Obično kreću od trenutnog stanja (nasuprot tome, ovde je prikazana analiza od konačnog stanja unazad) i izračunavaju moguće poteze koje oni i njihov protivnik može napraviti. Upotrebljavaju algoritme kao što je Minmax, koji promišlja moguće poteze računara pod pretpostavkom da će protivnik odigrati najbolji mogući potez. U složenijim igrama kao što je šah, računari analiziraju poteze do neke dubine (do oko 15 poteza unapred) i koriste aproksimativne metode za procenu pozicije.

Izaberi kamenje

Igraš igru sa dabrom. U ovoj igri postoji nekoliko kamena i igrači naizmenično biraju (uzimaju) kamenje. U svakom potezu igrač može izabrati 1,2 ili 3 kamena. Pobeđuje onaj igrač koji uzme poslednji kamen.

Ova igra se prvobitno igrala sa 9 kamena, a kroz istoriju se pokazalo da prvi igrač uvek može pobediti ako počne sa uzimanjem 1 kamena, bez obzira šta protivnik uradio sledeće. Sada, ti i dabar igrate sa 15 kamena.



Pitanje:

Ti počinješ igru. Koliko kamena treba da uzmeš da bi sigurno dobio igru?

- 1
- 2
- 3

Nema takve strategije koja bi dovela do toga da ti sigurno pobediš u igri.

Odgovor: 3

Informatička pozadina: ovo je zadatak iz teorije igara, koja je predmet proučavanja strateškog donošenja odluka. Primeri teorije igara mogu se naći u svakodnevnom životu, kao što su preduzeća koja se takmiče za udeo na tržištu ili političari koji se sučeljavaju na izborima. Teorija igara takođe se široko primenjuje u ekonomiji, biologiji, društvenim naukama i u računarstvu.

