

DABAR

MEĐUNARODNO NATJECANJE IZ INFORMATIKE I
RAČUNALNOG RAZMIŠLJANJA



2019.

ucitelji.hr

Uvod



Natjecanje iz informatike i računalnog razmišljanja – Dabar održano je u osnovnim i srednjim školama Republike Hrvatske od 11. do 15. studenoga 2019. Ovo je četvrta godina natjecanja, ali prvi puta na CARNet-ovom sustavu MOOC.

Natjecanje je besplatno za učenike i škole, a škole se moraju pridržavati Pravilnika natjecanja i Etičkog kodeksa natjecanja Dabar.

Stranice međunarodnog natjecanja su <http://bebras.org>, a hrvatsko sjedište je na portalu ucitelji.hr.

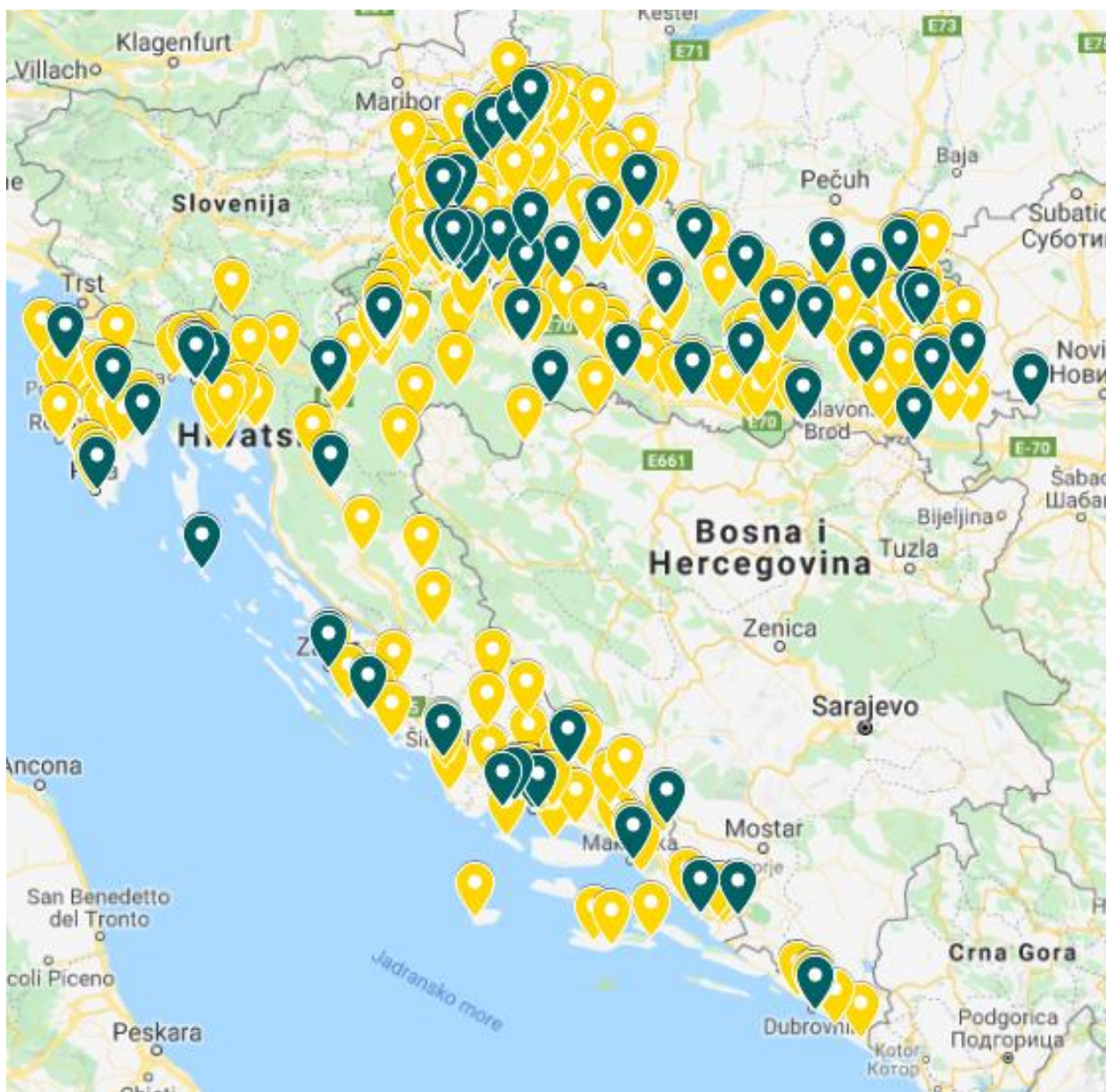
Organizator natjecanja za Hrvatsku je Udruga “Suradnici u učenju”. Podršku natjecanju i ove godine daju Hrvatski savez informatičara, Visoko učilište Algebra, CARNET i Ministarstva znanosti i obrazovanja.

Natjecanje ima pet kategorija, a učenici rješavaju 12 zadataka uz vremensko ograničenje od 40 minuta.

Iako je natjecanje održano u otežanim okolnostima zbog prekida održavanja nastave u nekim županijama u tjednu natjecanja, broj sudionika je nadmašio prošlogodišnji broj od 22887 učenika. Ove godine sudjelovalo je **24769** učenika.

Osobito nas veseli porast broja natjecatelja u kategoriji MikroDabar, dok je u odnosu na prethodnu godinu smanjen broj natjecatelja u kategoriji GigaDabar.

Na natjecanje se prijavilo [538 škola](#):



Broj učenika, po kategorijama je sljedeći:

- ✓ **MikroDabar** – 2042 učenika 1. i 2. razreda OŠ
- ✓ **MiliDabar** – 3563 učenika 3. i 4. razreda OŠ
- ✓ **KiloDabar** – 9466 učenika 5. i 6. razreda OŠ
- ✓ **MegaDabar** – 6582 učenika 7. i 8. razreda OŠ
- ✓ **GigaDabar** – 3116 učenika svih razreda SŠ

Maksimalni broj bodova u kategoriji MegaDabar je 11, dok su u svim ostalim kategorijama učenici ostvarili maksimalan broj bodova: 11 učenika u kategoriji MikroDabar, 2 učenika u kategoriji MiliDabar, 4 učenika u kategoriji KiloDabar i 16 učenika u kategoriji GigaDabar.

Najboljih 10% u svakoj kategoriji čine učenici koji su ostvarili sljedeći ili veći broj bodova:

- ✓ **MikroDabar – 8**
- ✓ **MiliDabar – 7**
- ✓ **KiloDabar – 7**
- ✓ **MegaDabar – 6,13**
- ✓ **GigaDabar – 8**

Za sve sudionike natjecanja pripremljene su digitalne značke i diplome koje samostalno preuzimaju iz sustava mooc.carnet.hr. Svi učenici dobili su i digitalne značke u Loomen-u. Oni najbolji (10% najboljih u svakoj kategoriji) dobili su i diplomu i značku za izniman uspjeh na natjecanju.

Natjecanje je i ovaj puta organizirano uz pomoć volontera koji čine organizacijski odbor natjecanja. Hvala im svima!

Hvala i svim učiteljima i nastavnicima koji su i ove godine organizirali natjecanje u svojim školama.

Hvala svim učenicima koji s osmijehom dolaze i odlaze s ovog natjecanja neovisno o postignutim rezultatima.

Hvala svima što ste prihvatili izazov!



Koordinatorica natjecanja Dabar 2019:

Darija Dasović

Kontakt e-mail adresa: dabar@ucitelji.hr

Programski odbor natjecanja Dabar 2019:

Darija Dasović, Sanja Pavlović Šijanović, Ela Veža, Vesna Tomić, Darko Rakić

Organizacijski odbor natjecanja Dabar 2019:

Jelena Nakić, Maristela Rubić, Marica Jurec, Mirela Radošević, Lidija Blagojević, Natalija Stjepanek, Sanda Šutalo, Sanja Križ, Ivana Bašić, Branka Čutura. Loredana Zima Krnelić, Renata Pintar, Anica Leventić, Daniela Usmiani, Barbara Knežević, Nikolina Bubica, Aleksandra Žufić, Valentina Pajdaković, Ines Kniewald, Tomislav Leček, Tanja Oreški, Maja Jurić-Babaj, Alma Šuto, Andrea Pavić. Iva Mihalić-Krčmar

5

U nastavku objavljujemo sve zadatke s natjecanja Dabar 2019. godine. Uz svaki zadatak naznačena je zemlja njegovog porijekla.

Svi sadržaji u ovoj Zbirci zadataka dostupni su pod Creative Commons licencom Imenovanje-Nekomercijalno-Dijeli pod istim uvjetima CC BY-NC-SA.



SADRŽAJ

Dimni signali	8
Čišćenje livade	9
Putovanje svemirom	10
Kutija s lopticama	11
Trgovina slatkiša	13
Papir za grebanje	14
Rođendanski kolač	16
Ukrašavanje	17
Kupalište	18
RAZNOBOJNI PUT	19
POSEBNI TORNJEVI	20
ZEC U ŠUMI	22
Raspored loptica	23
POPUNJAVANJE KUTIJA	24
PEČATI	25
Sjedala u kinu	27
Sef	29
Dabrov ples	31
Kokova farma	32
TAJNI JEZIK	33
Dimni signali	34
Put kući	36
ZAMJENA MAČAKA	37
Rangoli dizajn	38
Provala u školsku knjižnicu	40
Čišćenje snijega	41
DOSTAVA	42
Raspored sjedenja	43
Poruka starosjedilaca	44

Slavna osoba (zvijezda).....	46
Vrećica bombona	48
Prva i posljednja čarapa.....	49
Cinestar	50
Digitalni broj.....	52
Lažne vijesti	53
Kocka kocka kockica	54
Koji toranj?	55
Prikaz slike.....	56
Narukvice	58
Torte i susjede	61
Raspored sjedenja	62
Spremi u kutiju	63
Kineski znakovi	66
Stroj za pakiranje.....	68
Strojni vez.....	69
Crvenkapica	70
Recikliranje stakla.....	71
Pokretna traka.....	74
Video kompresija.....	76
Mostovi i otoci.....	77
Brojač	79
Zelenije linije letenja	81
Poruka starih dabrova	83
Znanstvenici u laboratoriju	85
Digitalni broj.....	86
Mreža dabrova	87
Šifrirana karta	88
Posjeta.....	90
Skladišta	92
Crtajuće trojke.....	93
Quipu.....	95
B-dabar kupuje bez ostatka	96

NARUKVICE

Oznaka zadatka: 2019-BE-04

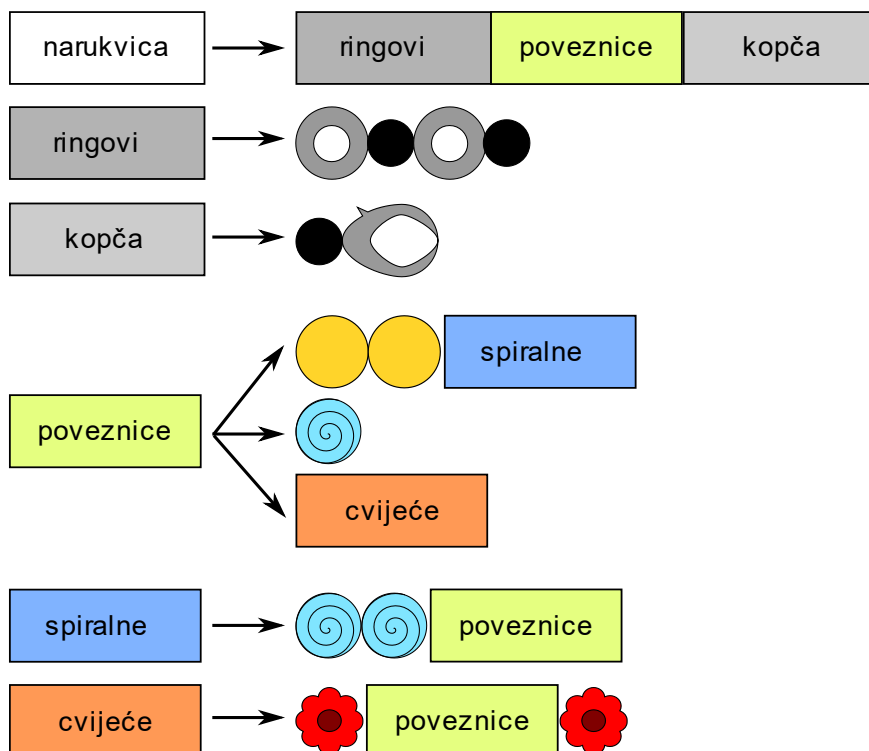
Tip pitanja: Višestruki odgovor

Ključne riječi: rekurzija, gramatika



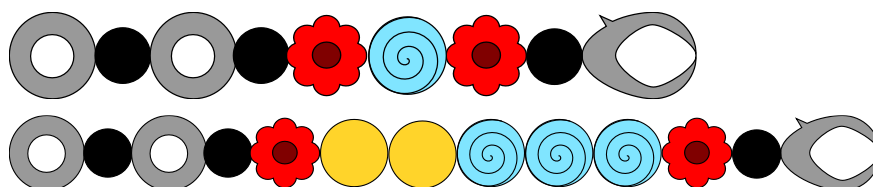
ZADATAK

Stipe izrađuje narukvice pridržavajući se sljedećih slikovnih pravila:



Odnosno, svaki simbol s lijeve strane zamijenjen je jednim od nizova simbola na koje upućuje s desne strane.

Na primjer, koristeći ova pravila nekoliko puta, Stipe može napraviti ove dvije narukvice.



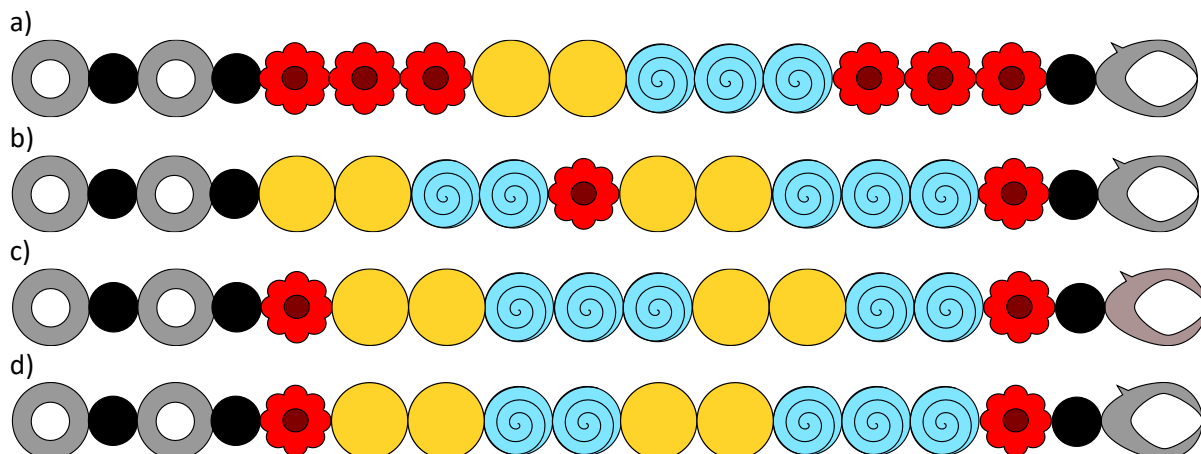
Stipe je napravio četiri narukvice za četvoricu svojih prijatelja, koristeći ista pravila. Jedan od prijatelja slomio je narukvicu i pogriješio pokušavajući to popraviti.

PITANJE/IZAZOV

Koja je od ove četiri narukvice bila slomljena?

PONUĐENI ODGOVORI

Odaberite jedan odgovor:

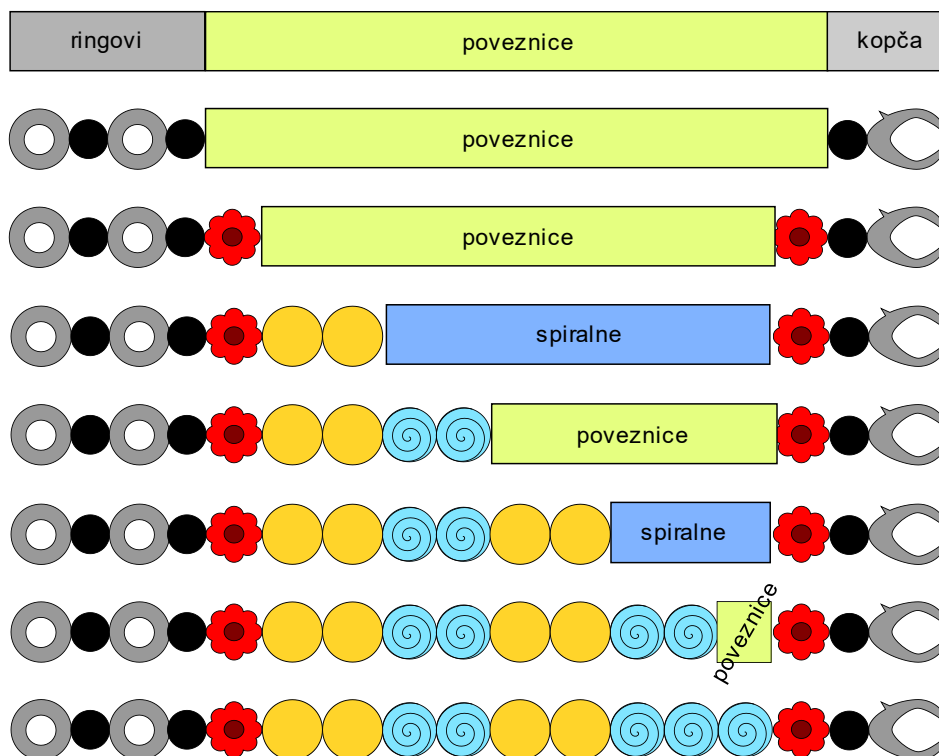


TOČAN ODGOVOR

Točan odgovor je C

OBJAŠNENJE

Umjesto dokazivanja da C ima pogrešku, možda bi bilo lakše pokazati da se ostale tri narukvice doista mogu napraviti pomoću ovih pravila. Za narukvicu D to prikazujemo na ovoj slici:

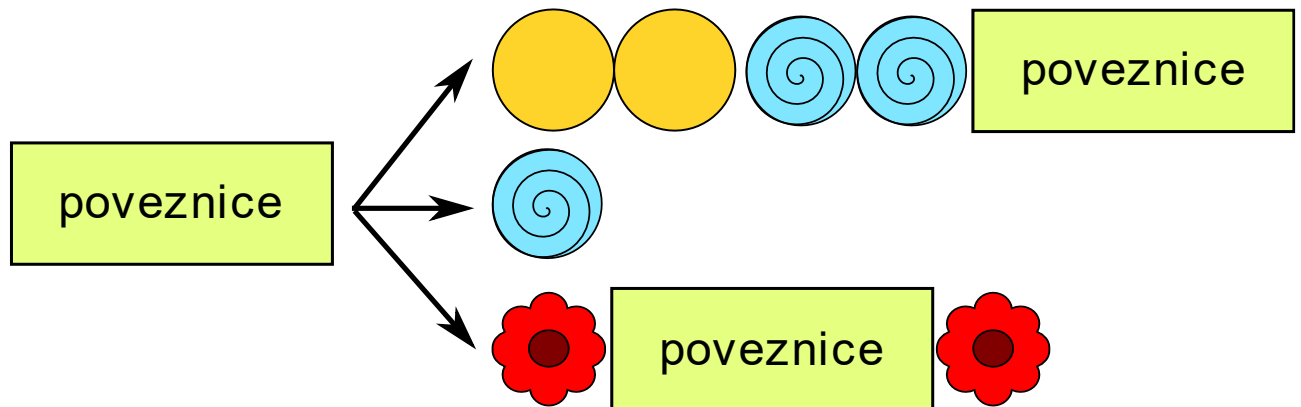


Nije tako teško napraviti sličnu sliku za A i B.

Da bismo dokazali da se C nikada ne može napraviti poštujući pravila, pogledajmo pravilo 'poveznice',

pravilo 'spiralne' i pravilo 'cvijeće'.

Zajedno se to može kombinirati u novo pravilo, koje ne mijenja vrstu narukvice koju je moguće proizvesti:



Drugim riječima, ako zanemarimo cvjetne kuglice, tada se svaka narukvica sastoji od uzorka ponovljenog više puta (moguće samo jednom, ili čak 0 puta), nakon čega slijedi pojedinačni uzorci.

Na narukvicama A, B i D jasno se vidi da slijede ovaj obrazac, dok narukvica C ima tri svijetloplave 'spiralne' poveznice u sredini narukvice, a to nije dopušteno.

RAČUNALNA POVEZANOST

Skup pravila koja su zadana u ovom zadatku računalo bi moglo koristiti za provjeru je li Stipe mogao napraviti određenu narukvicu. Računala koriste slična pravila za provjeru jesu li programeri u svojim programima napravili greške prilikom tipkanja ili za provjeru je li točno ono što unosite u web obrazac. Na primjer, sljedeća pravila

- broj → nepotpisano *ili* + nepotpisano *ili* - nepotpisano
- nepotpisano → znamenka *ili* nepotpisano znamenka
- znamenka → 0 *ili* 1 *ili* 2 *ili* 3 *ili* 4 *ili* 5 *ili* 6 *ili* 7 *ili* 8 *ili* 9

mogu računalu reći što je ispravan broj.

Takve skupove pravila, u računalnoj znanosti, nazivamo gramatikama, ili, u određenom kontekstu, slobodnim gramatikama.

TORTE I SUSJEDE

Oznaka zadatka: 2019-BE-05

Tip pitanja: Višestruki izbor

Ključne riječi: istodobnost (istovremenost)



ZADATAK

U Dabrogradu žive tri susjede: Ana, Branka i Klara. U petak ujutro svaka od njih je naručila u istoj slastičarnici po jednu tortu za subotnju zabavu. Sve su naručile jednaku vrstu torte visine 3 cm.

Međutim, svaka od njih je kasnije zvala slastičara kako bi izmijenila narudžbu. Slastičar je pritom zapisao novu narudžbu i bacio staru. Torte će se peći u subotu rano ujutro.

- Anin prvi poziv: ispecite moju tortu 1 cm višu od one koju sam naručila.
- Anin drugi poziv: ispecite moju tortu jednake visine kao što je Brankina.
- Brankin prvi poziv: ispecite moju tortu 2 cm višu od one koju sam naručila.
- Brankin drugi poziv: ispecite moju tortu 1 cm nižu od one koju sam naručila.
- Klarin prvi poziv: ispecite moju tortu 1 cm višu od Anine
- Klarin drugi poziv: ispecite moju tortu 1 cm višu od one koju sam naručila.



Napomena: Ne znamo u koje vrijeme su pozivi bili upućeni slastičaru. Znamo samo da je drugi poziv iste osobe bio upućen nakon prvog.

PITANJE/IZAZOV

Nakon što su u subotu sve tri torte ispečene, koja od sljedećih izjava je točna?

PONUĐENI ODGOVORI

- a) Anina i Brankina torta su jednake visine
- b) Brankina torta je barem 1 cm niža od Klarine torte
- c) Klarina torta je točno 2 cm viša od Anine
- d) Sve tri torte su visoke najmanje 4 cm

TOČAN ODGOVOR

b) Brankina torta je barem 1 cm niža od Klarine torte

OBJAŠNJENJE

Odgovor A nije točan – ako je Branka zvala tek nakon svih Aninih poziva, Ana će završiti s tortom visine 3 cm, a Branka s tortom visine 4 cm. To znači da niti odgovor D nije točan.

Odgovor C nije točan – ako je Klara ta koja je prva obavila svoje pozive, a Branka i Ana nakon nje, Klarina torta bit će visoka 5 cm, a Brankina i Anina 4 cm.

Ako se pogledaju Brankine upute slastičaru, vidjet će se da je ona naručila najprije tortu visine 3 cm, nakon toga 5 cm, a na kraju 4 cm. Posljedično, jedine vrijednosti koje je slastičar mogao zapisati za Aninu tortu su 3 cm, 4 cm ili 5 cm. Klarina torta može biti 5 cm, 6 cm ili 7 cm visoka.

Budući da će Brankina torta uvijek biti visoka 4 cm, samo odgovor B je sigurno točan.

RAČUNALNA POVEZANOST

Računalo može istovremeno raditi mnogo različitih stvari, čak i unutar istog programa. Na primjer, u programu za obradu teksta alat za provjeru pravopisa radi dok tipkate.

Kao što se iz zadatka vidi, kada se istovremeno izvršavaju različiti zadaci, nije uvijek lako predvidjeti kakav će biti konačni ishod, jer je ponekad teško predvidjeti kada će se točno izvršiti svaki dio svakog zadatka.

Pisanje programa koji se sastoje od zadataka koji se istodobno izvode naziva se paralelno programiranje. Programeri moraju biti svjesni problema koji sudjeluju u ovoj vrsti programiranja i pisati programe tako da se određene akcije uvijek izvode istim redoslijedom. Na primjer, programeri koji pišu programe za mrežni pisač moraju osigurati da se dva dokumenta koja su poslana na ispisivanje (otprilike) istovremeno, ispišu jedan za drugim. Program mora osigurati da se stranice različitih dokumenata ne izmiješaju ili da se na istoj stranici ne ispišu dijelovi različitih dokumenata. Današnja računala mogu raditi mnogo stvari istovremeno, stoga paralelno programiranje u posljednje vrijeme dobiva veliku važnost u nastavnom programu informatike.

RASPORED SJEDENJA

Oznaka zadatka: 2019-CA-05b

Tip pitanja: Odabir riječi koje nedostaju

Ključne riječi: logika

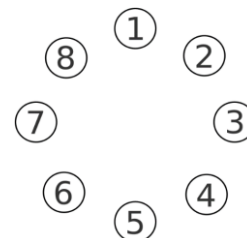


ZADATAK

Osam prijatelja sjedi u krugu, kao što je prikazano na slici. Svi gledaju prema sredini kruga. Stolicice su označene brojevima od 1 do 8 u smjeru kazaljke na satu.

Poznate su nam sljedeće činjenice vezano uz mjesta gdje prijatelji sjede:

1. Ana sjedi na stolici označenom brojem 1, točno nasuprot Domagoju.
2. Hrvoje sjedi između Gorane i Eve.
3. Fran ne sjedi pokraj Ane ili Domagoja.
4. Jedna osoba sjedi između Gorane i Cvijete.
5. Eva sjedi pokraj Domagoja, s njegove lijeve strane.

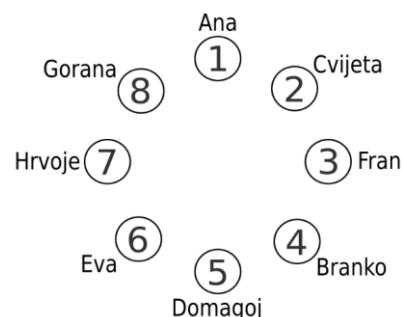


PITANJE/IZAZOV

Ispravno smjesti imena prijatelja za stolom na njihovo mjesto označeno brojem, gledano u smjeru kazaljke na satu.

TOČAN ODGOVOR

Ana (1), Cvijeta (2), Fran (3), Branko (4), Domagoj (5), Eva (6), Hrvoje (7), Gorana (8).



OBJAŠNENJE

Činjenica broj 1 „Ana sjedi točno nasuprot Domagoju“ omogućuje nam da za stol smjestimo Domagoja.

Činjenica broj 5 „Eva sjedi pokraj Domagoja, s njegove lijeve strane.“ omogućuje nam da smjestimo Evu. Sada se može iskoristiti činjenica broj 2 „Hrvoje sjedi između Gorane i Eve“ da saznamo gdje sjedi Hrvoje, a kada to znamo, znamo i gdje sjedi Gorana.

Do ovog trenutka već je petero prijatelja smješteno za stol, a činjenica broj 3 „Fran ne sjedi pokraj Ane ili Domagoja“ otvara nam mjesto za smještaj Frana.

Činjenica broj 4 „Jedna osoba sjedi između Gorane i Cvijete“ pokazuje nam mjesto na kojem sjedi Cvijeta. I konačno, ostalo je jedno mjesto slobodno i na njega se smješta Branko.

RAČUNALNA POVEZANOST

U logici je potrebno slijediti pravila, razumjeti redoslijed obavljanja operacija i negaciju.

Na primjer, ključni logički operator koji se koristi u ovom zadatku je negacija, koju koristimo uvijek kada kažemo „ne“.

Činjenicu „Fran ne sjedi pokraj Ane ili Domagoja“ možemo izreći i kao „Fran ne sjedi pored (Ane ili Domagoja)“ što je isto kao „(Fran ne sjedi pored Ane) i (Fran ne sjedi pored Domagoja)“.

Operator negacije mijenja logički izraz $ne(A \text{ ili } B)$ u $(ne A) \text{ i } (ne B)$, što je primjer DeMorganovog zakona. Logičke operacije mogu se prepravljati, kombinirati ili pojednostavljivati, što je vrlo korisno u informatici.

SPREMI U KUTIJU

Oznaka zadatka: 2019-CA-06

Tip pitanja: točan odgovor

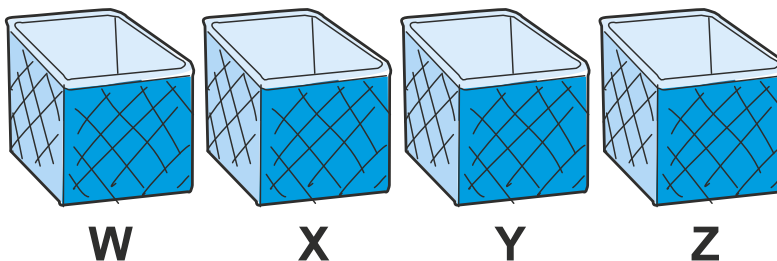
Ključne riječi: logičke funkcije



ZADATAK

Mama Dabrica ima četiri igračke koje želi staviti u četiri kutije označene slovima X, Y, Z i W. U svaku kutiju može staviti samo jednu igračku. Pritom **svi sljedeći uvjeti** moraju biti **istiniti**:

1. Igračka **je** u kutiji X. **ILI** Igračka **nije** u kutiji Y. **ILI** Igračka **nije** u kutiji Z.
2. Igračka **je** u kutiji W. **ILI** Igračka **je** u kutiji X. **ILI** Igračka **nije** u kutiji Z.
3. Igračka **nije** u kutiji X. **ILI** Igračka **nije** u kutiji Y. **ILI** Igračka **je** u kutiji Z.
4. Igračka **nije** u kutiji W. **ILI** Igračka **nije** u kutiji X. **ILI** Igračka **nije** u kutiji Y.
5. Igračka **nije** u kutiji X. **ILI** Igračka **je** u kutiji Y. **ILI** Igračka **nije** u kutiji Z.



63

PITANJE/IZAZOV

Koliko najviše igračaka Mama Dabrica može staviti u kutije tako da svi uvjeti budu istiniti? Kao odgovor upiši samo broj!

TOČAN ODGOVOR

3

OBJAŠNJENJE

U kutije se može staviti najviše tri igračke. Kad bi se stavilo sve četiri igračke u kutije (u svaku kutiju po jedna igračka) tada uvjet 4. ne bi bio istinit (po njemu mora barem jedna od kutija X, Y ili W biti prazna). Svi logički uvjeti imaju po tri logičke izjave povezane logičkom funkcijom **ILI**. Da bi uvjet oblika A **ILI** B **ILI** C bio istinit bar jedna od izjava A, B ili C mora biti istinita.

Uvjete možemo prikazati tablicom ako izjavu „igračka je u kutiji“ označimo s **1**, a izjavu „igračka nije u kutiji“ označimo s **0**:

uvjet	W	X	Y	Z
1	-	1	0	0
2	1	1	-	0
3	-	0	0	1
4	0	0	0	-
5	-	0	1	0

Budući da tražimo najveći broj igračaka koji se može staviti u kutije, provjeravamo sve uvjete redom počevši od 4 igračke:

- 1) Četiri igračke ne mogu se staviti u kutije, jer bi tada logički uvjet 4) bio lažan: da bi bio istinit barem jedna od kutija X, Y ili W mora biti prazna.
- 2) Ako stavimo tri igračke u kutije, tada je jedna od kutija prazna:

a) Ako je prazna kutija W, tada jesu ispunjeni svi uvjeti:

- 1) Igračka je u X pa je uvjet istinit.
- 2) Igračka je u X pa je uvjet istinit.
- 3) Igračka je u Z pa je uvjet istinit.
- 4) Igračka nije u W pa je uvjet istinit.
- 5) Igračka je u Y pa je uvjet istinit.

b) Ako je prazna kutija X, tada nisu ispunjeni svi uvjeti:

- 1) Igračka nije u X, ali je u Y i Z pa je uvjet **lažan**.
- 2) Igračka je u W pa je uvjet istinit.
- 3) Igračka je u Z pa je uvjet istinit.
- 4) Igračka nije u X i je u Z pa je uvjet istinit.
- 5) Igračka je u Y i nije u X pa je uvjet istinit.

c) Ako je prazna kutija Y, tada nisu ispunjeni svi uvjeti:

- 1) Igračka nije u Y i je u X pa je uvjet istinit.
- 2) Igračka je u W i u X pa je uvjet istinit.
- 3) Igračka je u Z pa je uvjet istinit.
- 4) Igračka nije u Y pa je uvjet istinit.
- 5) Igračka je u Y i nije u X i Z pa je uvjet **lažan**.

d) Ako je prazna kutija Z, tada nisu ispunjeni svi uvjeti:

- 1) Igračka je u X i nije u Z pa je uvjet istinit.
- 2) Igračka je u X i W i nije u Z pa je uvjet istinit.
- 3) Igračka nije u Z i je u X i u Y pa je uvjet **lažan**.
- 4) Igračka nije u Z pa je uvjet istinit.
- 5) Igračka je u Y i nije u Z pa je uvjet istinit.

Znači, jedini način da Mama Dabrica stavi 3 igračke u kutije je da ih stavi u kutije X, Y i Z, a kutiju W ostavi praznu.

RAČUNALNA POVEZANOST

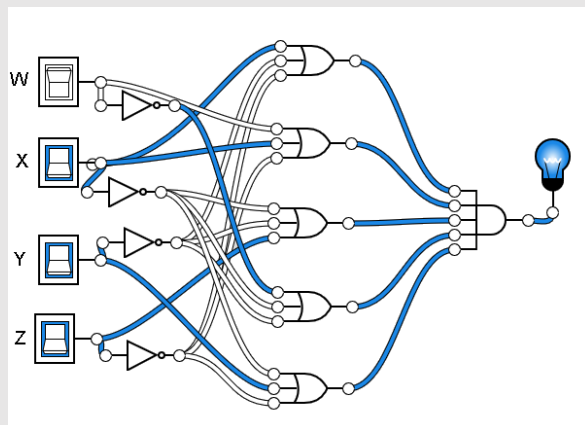
Logički uvjeti vrlo su česti u informatici. Radi se o problemu zadovoljavanja više uvjeta. Ovih 5 logičkih izjava mogli smo povezati u jednu složenu izjavu logičkom funkcijom I. Ako s **X** označimo istinitu izjavu, a s $\bar{X}$ lažnu onda možemo pisati:

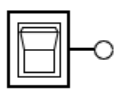
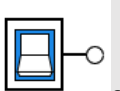
$(X \text{ ILI } \bar{Y} \text{ ILI } \bar{Z}) \text{ I } (W \text{ ILI } X \text{ ILI } \bar{Z}) \text{ I } (\bar{X} \text{ ILI } \bar{Y} \text{ ILI } Z) \text{ I } (\bar{W} \text{ ILI } \bar{X} \text{ ILI } Y) \text{ I } (\bar{X} \text{ ILI } Y \text{ ILI } \bar{Z})$. U matematici i računalstvu često se **I** piše kao $\wedge$, a **ILII** kao $\vee$ pa izjavu pišemo ovako:

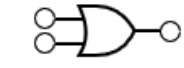
$$(X + \bar{Y} + \bar{Z}) (W + X + \bar{Z}) (\bar{X} + \bar{Y} + Z) (\bar{W} + \bar{X} + Y) (\bar{X} + Y + \bar{Z})$$

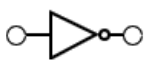
$$(X \vee \bar{Y} \vee \bar{Z}) \wedge (W \vee X \vee \bar{Z}) \wedge (\bar{X} \vee \bar{Y} \vee Z) \wedge (\bar{W} \vee \bar{X} \vee Y) \wedge (\bar{X} \vee Y \vee \bar{Z})$$

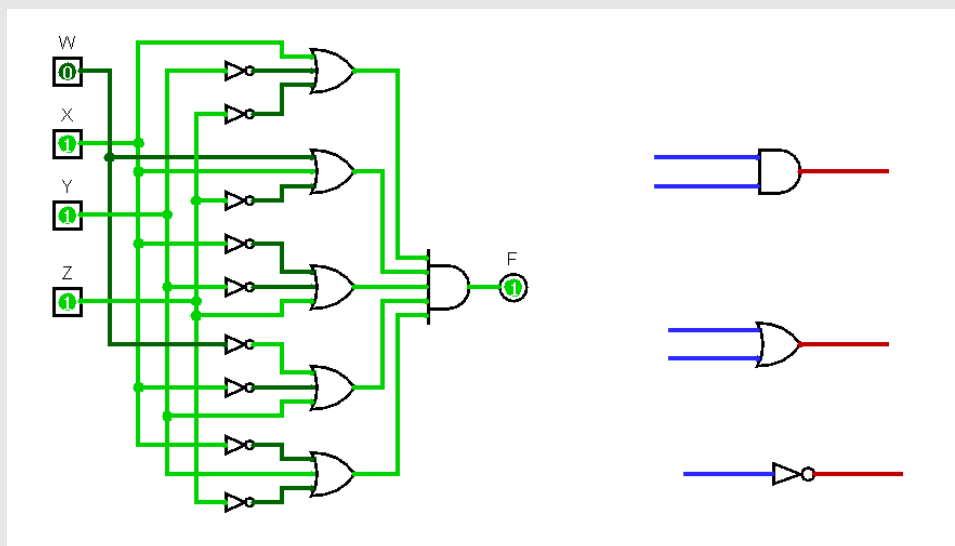
Grafički ovu složenu logičku izjavu možemo prikazati pomoću logičkog sklopa (<https://logic.ly/demo>). Logička izjava je istinita samo ako je W lažan, a X, Y i Z istiniti:



Sklopka  označava 0 ili *laž*, a sklopka  označava 1 ili *istina*. Sklop I je 

, a sklop ILI je . Oba sklopa mogu imati 2 ili više ulaza. Sklop I daje istinit izlaz, samo ako su svi ulazi istiniti (plava boja), a sklop ILI daje istinit izlaz, ako je barem jedan uvjet istinit.

Sklop NE prikazan je simbolom  te uvijek ima samo jedan ulaz i samo jedan izlaz. On pretvara *istinu* u *laž*, odnosno *laž* u *istinu*. Također, na slici su bijele žice niskog napona (*lažna* izjava), a plave žice visokog napona (*istinita* izjava).



Inače, ovaj problem je poznat kao "Boolean satisfiability problem" (kratica: SAT), u prijevodu: „problem koji zadovoljava pravila Boolove algebre“. Kad se poveća broj uvjeta, problem može biti vrlo složen za rješavanje. Mi imamo samo 5 uvjeta, ali ih treba svaki puta ponovno provjeriti kad mijenjamo početnu vrijednost bilo koje logičke varijable (konkretno: ima li igračke u kutiji ili ne). Informatičari poznaju bolje strategije, ali sve su prilično neučinkovite. Postoje SAT natjecanja na kojima programeri pokušavaju napisati najbrži "SAT solver" (SAT rješavač). Budući da se mnogi različiti problemi mogu preslikati na ekvivalentni SAT, SAT rješenja mogu biti korisna u mnogim situacijama.

KINESKI ZNAKOVI

Oznaka zadatka: 2019-CN-03a

Tip pitanja: Višestruki odgovor

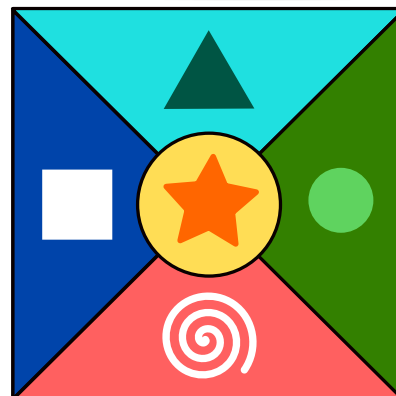
Ključne riječi: prikaz, struktura-obrazac



ZADATAK

Kineski znakovi privlače pozornost malog dabra koji namjerava proučiti njihovu strukturu. Dabar stvara strukturnu ploču ispunjenu bojom i uzorkom, što je prikazano na ovoj slici:

Prema ovoj ploči sljedeći kineski znakovi mogu biti izraženi kao:



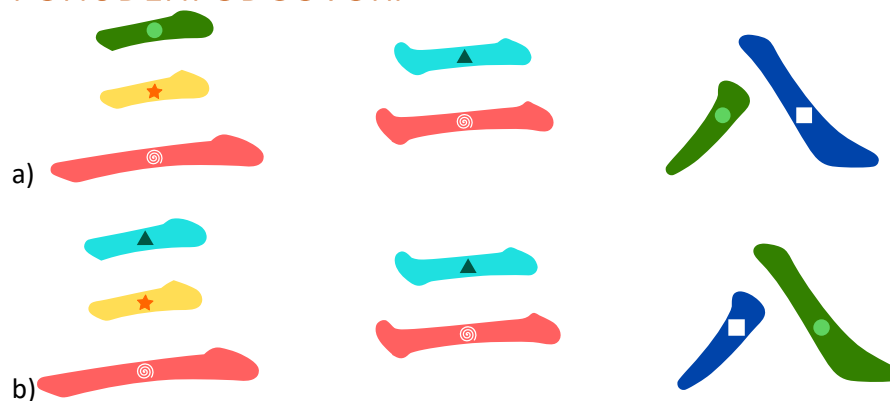
"川"	"儿"	"吕"
Kao lijeva-srednja-desna struktura	Kao lijeva-desna struktura	Kao gornja-donja struktura

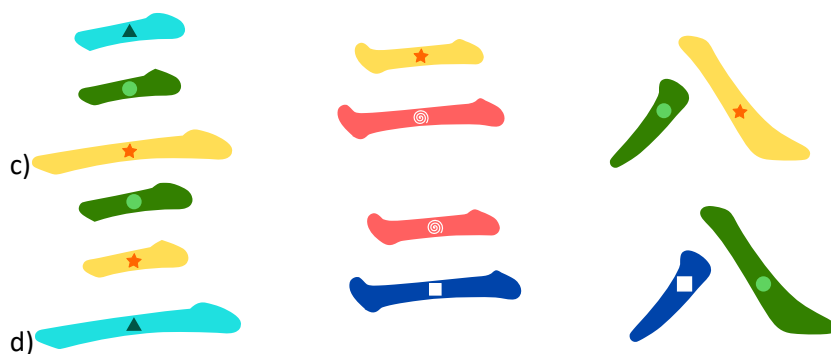
66

PITANJE/IZAZOV

Koja ponuđena opcija predstavlja znakove "三", "二", "八" na točan način ?

PONUĐENI ODGOVORI





TOČAN ODGOVOR

Točan odgovor je b)



OBJAŠNJENJE

Znak "三" je struktura gornji-srednji-donji, pa je gornji dio svijetloplave boje , srednji dio je žut ; donji dio je ružičaste boje .

Znak "二" je gornja-donja struktura, pa bi gornji dio trebao biti svijetloplave boje , donji dio ružičaste boje .

Znak "八" je lijevo-desna struktura, pa je lijevi dio plave boje , a desni dio zelene boje .

Dakle, ispravna opcija je B.

Opcija A: Znak "二" predstavlja ispravno rješenje, ali "三" i "八" predstavljaju pogrešno rješenje.

Opcija C: Svi prikazi predstavljaju pogrešno rješenje.

Opcija D: Znak "八" predstavlja ispravno rješenje, ali "二" i "三" predstavljaju pogrešno rješenje.

RAČUNALNA POVEZANOST

Podaci, strukture podataka i prikazivanje:

Podaci mogu imati različite oblike, primjerice, slike, tekst ili brojeve. Kada pogledamo podatke u ovom pitanju, tražimo redoslijed slika koji će nam pomoći u rješavanju problema. Identificiranjem ovih slika možemo predvidjeti, stvarati pravila i rješavati opće probleme.

Kineski znakovi imaju strukturu sastavljenu od dijelova poput "građevnih blokova" koji su raspoređeni u dvodimenzionalnu strukturu. Ti se blokovi uglavnom pojavljuju u oblicima lijevo-desno, gore-dolje tvoreći dvodimenzionalni znak. Više: <http://www.littlechinesechannel.com/>

STROJ ZA PAKIRANJE

Oznaka zadatka: 2019-JP-05

Tip pitanja: Kratki odgovor

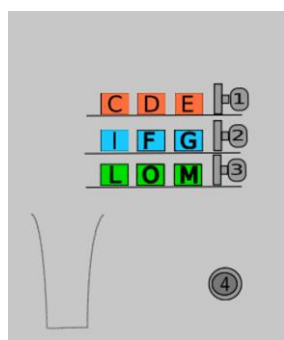
Ključne riječi: stog



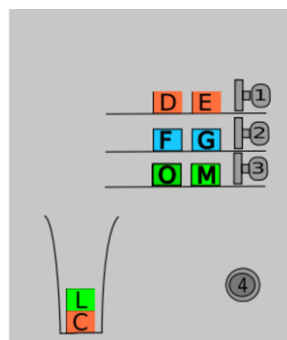
ZADATAK

Stroj pakira kutije označene slovima u vreću pritiskom na numeričke tipke i to prema ovom pravilu: kad se pritisne tipka 1, 2 ili 3, onda stroj krajnje lijevu kutiju s police pod odgovarajućim brojem stavlja u vreću, a kada se pritisne tipka 4, stroj ukloni najgornju kutiju iz vreće.

Na primjer, kada se pritisnu numeričke tipke u redoslijedu 1, 2, 4 i 3 sa slike 1, tada se ukloni kutija označena slovom I, a kutija označena slovom C ostaju u vreći, pa se potom u vreću ispusti kutija označena slovom L kako je prikazano na slici 2.



Slika 1



Slika 2

PITANJE/IZAZOV

Kada se pritisnu numeričke tipke sa Slike 1 redoslijedom 3, 1, 1, 4, 4, 2, 3 i 4, koja će se kutija nalaziti na drugom mjestu od vrha vreće? Upišite samo slovo kojim je kutija označena.

TOČAN ODGOVOR

Kutija L će biti na drugom mjestu od vrha vreće.

OBJAŠNJENJE

1. Stroj za pakiranje ispušta kutije u ovom redoslijedu L, C, D, pa je tako kutija D na vrhu, ispod nje kutija C, a na dnu je kutija L.
2. Stroj tada uklanja dvije kutije koja su na vrhu, a to su kutije D i C, pa u vreći ostaje samo kutija L.
3. Stroj sada ispušta kutije I i O. Sada je kutija O na vrhu, kutija I ispod kutije O, a kutija L na dnu vreće.
4. Stroj uklanja kutiju O koja je na vrhu. Pa u vreći ostaju kutija I na vrhu, a kutija L ispod kutije I.

Stoga su kutije, gledano odozgo, složene u redoslijedu I i L, pa se tako kutija L nalazi na drugom mjestu od vrha u vreći.

RAČUNALNA POVEZANOST

Posljednja kutija stavljena u vreću može se prva izvaditi. Takva struktura naziva se **stog** (*stack*, apstraktni tip podataka). Računala to koriste u manipulaciji podacima kada programeri žele dobiti podatke prema protokolu „Zadnji ulazi prvi izlazi“ (Last In First Out, LIFO). To je korisno, na primjer, za provjeru jesu li zagrade ispravno uparene, jer je prva zagrada koja se mora zatvoriti posljednja zagrada koja je otvorena. Postoji i struktura podataka koja se zove **red** (*queue*).

STROJNI VEZ

Oznaka zadatka: 2019-KR-05

Tip pitanja: Višestruki izbor

Ključne riječi: naredbe



ZADATAK

Dabar želi napraviti program kojim će stroj za vezenje moći vesti različite uzorke. Za izradu programa dabar koristi naredbu **OUT(cc)-IN(dd)**, pri čemu **cc** i **dd** označavaju položaj igle u koordinatnoj mreži.

Na primjer, naredba **OUT(B2)-IN(A3)** će pomaknuti iglu na položaj B2, probosti platno i provući konac od ispod prema gore, a nakon toga se pomaknuti na položaj A3, probosti platno i provući konac odozgo prema dolje.

Za izradu obrasca prikazanog na slici potrebne su naredbe **OUT(E6)-IN(G8);OUT(E2)-IN(E4)**.

	1	2	3	4	5	6	7	8	9	10
A	○	○	○	○	○	○	○	○	○	○
B	○	○	○	○	○	○	○	○	○	○
C	○	○	○	○	○	○	○	○	○	○
D	○	○	○	○	○	○	○	○	○	○
E	○	—	—	○	○	○	○	○	○	○
F	○	○	○	○	○	○	○	○	○	○
G	○	○	○	○	○	○	○	○	○	○
H	○	○	○	○	○	○	○	○	○	○
I	○	○	○	○	○	○	○	○	○	○
J	○	○	○	○	○	○	○	○	○	○

PITANJE/IZAZOV

Koje naredbe će stvoriti uzorak mašne na slici?

PONUĐENI ODGOVORI

- a) **OUT(H2)-IN(C2);OUT(H9)-IN(C9);OUT(C9)-IN(C2);OUT(H9)-IN(C2)**
- b) **OUT(C2)-IN(H9);OUT(H2)-IN(C9);OUT(C2)-IN(H2);OUT(C9)-IN(H9)**
- c) **OUT(H9)-IN(C9);OUT(H9)-IN(H2);OUT(C2)-IN(H2);OUT(C9)-IN(H2)**
- d) **OUT(C2)-IN(C9);OUT(H2)-IN(H9);OUT(C2)-IN(H2);OUT(C9)-IN(H9)**

TOČAN ODGOVOR

B) **OUT(C2)-IN(H9);OUT(H2)-IN(C9);OUT(C2)-IN(H2);OUT(C9)-IN(H9)**

	1	2	3	4	5	6	7	8	9	10
A	○	○	○	○	○	○	○	○	○	○
B	○	○	○	○	○	○	○	○	○	○
C	○	○	○	○	○	○	○	○	○	○
D	○	○	○	○	○	○	○	○	○	○
E	○	○	○	○	○	○	○	○	○	○
F	○	○	○	○	○	○	○	○	○	○
G	○	○	○	○	○	○	○	○	○	○
H	○	○	○	○	○	○	○	○	○	○
I	○	○	○	○	○	○	○	○	○	○
J	○	○	○	○	○	○	○	○	○	○

OBJAŠNJENJE

Da bi se stvorio zadani uzorak, potrebne su 4 naredbe, po jedna za svaku liniju veza, pri čemu redoslijed vezenja nije važan. Te naredbe su:

- **OUT(C2)-IN(H9)** ili **OUT(H9)-IN(C2)**
- **OUT(H2)-IN(C9)** ili **OUT(C9)-IN(H2)**
- **OUT(C2)-IN(H2)** ili **OUT(H2)-IN(C2)**
- **OUT(C9)-IN(H9)** ili **OUT(H9)-IN(C9)**

Od ponuđenih odgovora, odgovor B sadrži sve potrebne naredbe.

U odgovoru A nalazi se naredba **OUT(C9)-IN(C2)** koja veze liniju koja nije potrebna, dok istovremeno nedostaje naredba **OUT(H2)-IN(C9)** ili **OUT(C9)-IN(H2)**.

Odgovor B je točan.

U odgovoru C nalazi se naredba **OUT(H9)-IN(H2)** koja veze nepotrebnu liniju, a nedostaje naredba **OUT(C9)-IN(H9)** ili **OUT(H9)-IN(C9)**.

U odgovoru D nalaze se naredbe **OUT(C2)-IN(C9)** i **OUT(H2)-IN(H9)** koje vezu dvije nepotrebne linije, a nedostaju naredbe **OUT(C2)-IN(H9)** ili **OUT(H9)-IN(C2)** i naredba **OUT(H2)-IN(C9)** ili **OUT(C9)-IN(H2)**.

RAČUNALNA POVEZANOST

Algoritam opisuje korake koje treba slijediti kako bi se izvršio neki zadatak. Algoritmi su uobičajeni u informatici, ali se koriste i za rješavanje problema u svakodnevnom životu. Ovaj zadatak je primjer kako se algoritam može koristiti za stvaranje uzorka za vezenje.

CRVENKAPICA

Oznaka zadatka: 2019-RO-02

Tip pitanja: višestruki odgovor

Ključne riječi: dinamično programiranje, dvodimenzionalni nizovi



ZADATAK



Crvenkapica želi ubrati cvijeće iz bakinog vrta. Vrt je podijeljen u više dijelova, a u svakom dijelu zasađen je određeni broj cvjetova. Crvenkapica započinje s branjem cvijeća u gornjem lijevom dijelu vrta, a završava u donjem desnom dijelu vrta, pri čemu se kreće samo prema dolje ili desno.

PITANJE/IZAZOV

Koliko najviše cvjetova Crvenkapica može ubrati za vrijeme šetnje vrtom, na opisani način ?

PONUĐENI ODGOVORI

a) 35

b) 38

c) 58

d) 41

TOČAN ODGOVOR

Točan odgovor je D) 41

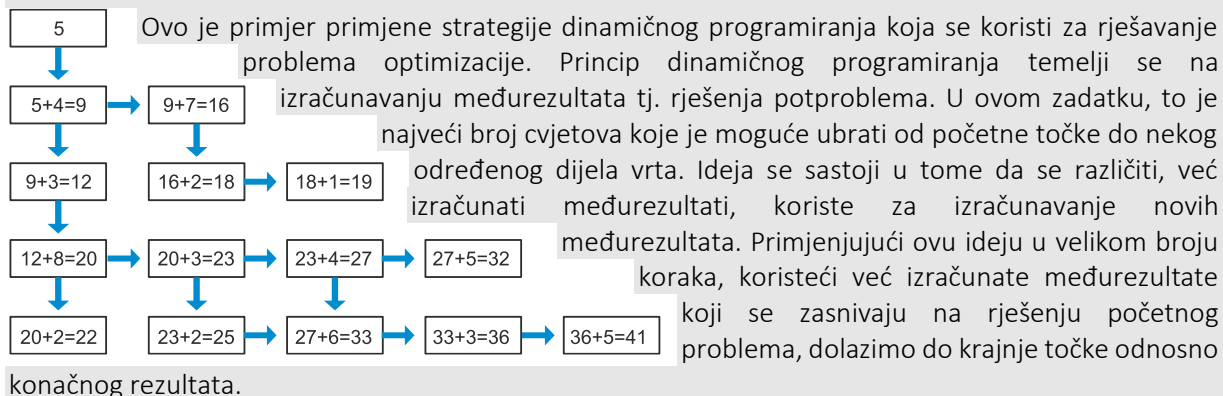
OBJAŠNENJE

Za svaki dio vrta moguće je izračunati najveći mogući broj cvjetova koje Crvenkapica može ubrati od početka do tog dijela vrta (nazovimo taj broj *Maksimum*). *Maksimum* se može izračunati na sljedeći način:

Ako se do određenog dijela vrta može doći samo iz jednog od prethodnih dijelova vrta, onda je *Maksimum* jednak zbroju cvjetova u tom dijelu i *Maksimuma* prethodnog dijela vrta.

Ako se do određenog dijela vrta može doći iz dva prethodna dijela vrta, onda je *Maksimum* jednak zbroju cvjetova u tom dijelu vrta i većem od dva *Maksimuma* prethodnih dijelova.

RAČUNALNA POVEZANOST



Mnogi praktični problemi optimizacije mogu se riješiti ovom strategijom koja često predstavlja vremenski vrlo učinkovit algoritam. https://en.wikipedia.org/wiki/Dynamic_programming

RECIKLIRANJE STAKLA

Oznaka zadatka: 2019-RO-03

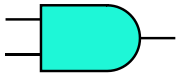
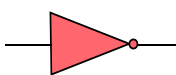
Tip pitanja: višestruki odgovori

Ključne riječi: ogički sklopovi, logičke funkcije

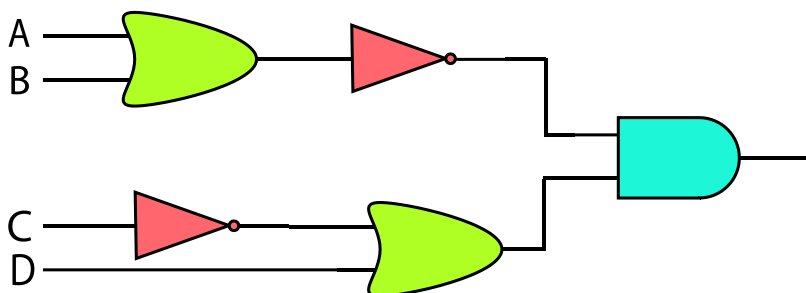


ZADATAK

U Nelinoj školi puno pažnje polaže se recikliranju (oporabi) stakla. Staklo je bijelo ili obojeno. Ravnateljica Nela dobila je magične strojeve za uporabu (reciklažu) stakla. Dvije vrste strojeva uzimaju po dva komada stakla i pretvaraju ih u novo staklo. Treći stroj uzima samo jedan komad stakla i pretvara ga u novo staklo. Pretvorbe se izvršavaju na sljedeći način:

	Ovaj stroj će proizvesti bijelo staklo samo ako su u stroj stavljena dva komada bijelog stakla. Stavljanjem bilo koje druge kombinacije boja stroj će proizvesti obojeno staklo.
	Ovaj stroj će proizvesti obojeno staklo samo ako su u stroj stavljena dva komada obojenog stakla. Stavljanjem bilo koje druge kombinacije boja stroj će proizvesti bijelo staklo.
	Ovaj stroj će od bijelog stakla proizvesti obojeno, a od obojenog stakla proizvesti bijelo.

Ravnateljica je napravila sljedeći sustav:



71

PITANJE/IZAZOV

Koje vrste stakla mogu biti stavljene u strojeve na ulaze A, B, C i D, tako da sustav proizvede bijelo staklo?

PONUĐENI ODGOVORI

- a) A=bijelo, B= bijelo, C= obojeno, D=bijelo
- b) A= obojeno, B= obojeno, C= obojeno, D=bijelo
- c) A= bijelo, B= obojeno, C= obojeno, D=bijelo
- d) A= obojeno, B= obojeno, C= bijelo, D=obojeno

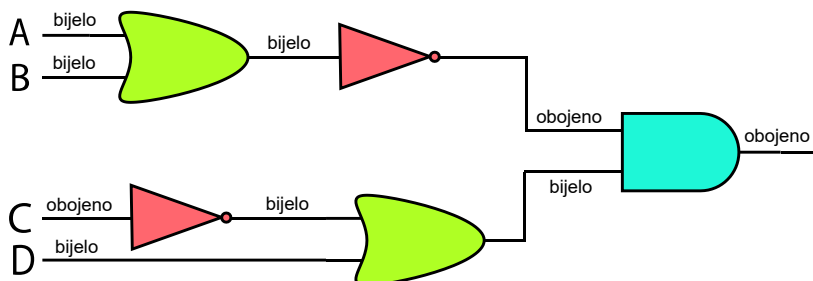
TOČAN ODGOVOR

Točan odgovor je b) A= obojeno, B= obojeno, C= obojeno, D=bijelo.

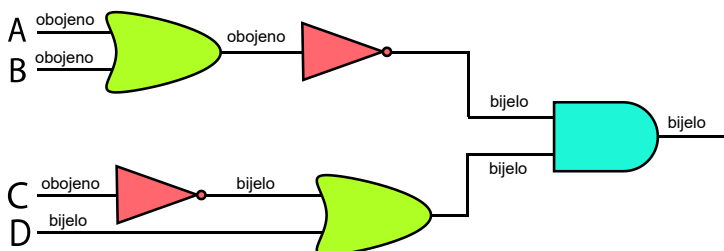
OBJAŠNJENJE

Nacrtajmo sliku za svako ponuđeno rješenje i iznad svake crte (ulazne i izlazne) napišimo koju vrstu stakla predstavlja. Na taj način moguće je utvrditi što je izlaz za svako od ponuđenih rješenja i pronaći točan odgovor. Jedina slika koja na izlazu daje bijelo staklo je Slika 2.

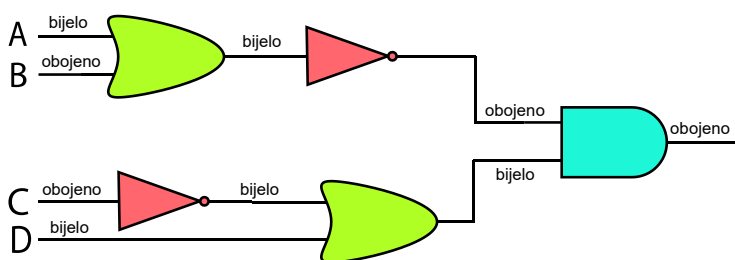
Slika 1



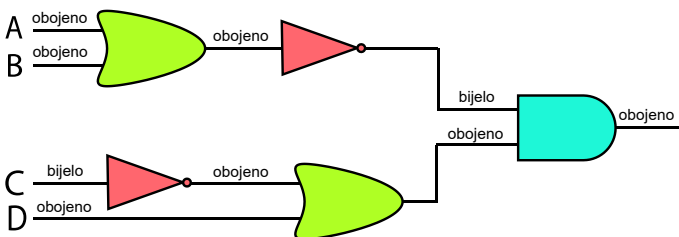
Slika 2



Slika 3



Slika 4



Uoči da postoje tri različite kombinacije ulaznih vrijednosti koje na izlazu daju bijelo staklo:

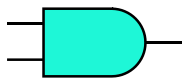
1. A= obojeno, B= obojeno, C= obojeno, D=obojeno
2. A= obojeno, B= obojeno, C= obojeno, D=bijelo (to je gore ponuđeni odgovor)
3. A= obojeno, B= obojeno, C= bijelo, D=bijelo

RAČUNALNA POVEZANOST

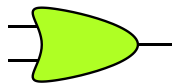
Računala su građena od složenih sklopova koji se sastoje od različitih vrsta sićušnih elemenata koje nazivamo logički sklopovi ili vrata. Neki od popularnih logičkih sklopova su NE, ILI, I, XILI (NOT, OR, AND, XOR).

U ovom zadatku možemo uočiti I, ILI i NE (AND, OR i NOT) vrata (njihov grafički prikaz u zadatku istovjetan je prikazu u tehničkoj literaturi).

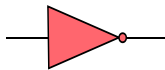
Za svoj rad logički sklopovi koriste električne signale. Postojanje signala označavamo s 1, a odsustvo signala označavamo s 0. U logici, signal 1 predstavlja ISTINU (TRUE), a signal 0 predstavlja LAŽ (FALSE). U ovom zadatku bijelo staklo predstavlja ISTINU (ili 1), a obojeno staklo LAŽ (ili 0).



I logički sklop zahtijeva da oba ulaza budu ISTINA (bijelo staklo) da bi izlaz bio ISTINA.



ILI logički sklop zahtijeva da barem jedan ulaz bude ISTINA da bi izlaz bio ISTINA.



NOT logički sklop vraća suprotnu vrijednost, što znači, ako je na ulazu ISTINA, na izlazu je LAŽ, odnosno, ako je na ulazu LAŽ, na izlazu je ISTINA.

Logički sklop (logička vrata) koristi se, između ostalog, u arhitekturi mikroprocesora za izvođenje aritmetičkih i logičkih operacija.

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

https://hr.wikipedia.org/wiki/Logički_sklopovi

POKRETNNA TRAKA

Oznaka zadatka: 2019-RS-06

Tip pitanja: višestruki izbori

Ključne riječi: programiranje, IF, IF THEN, WHILE

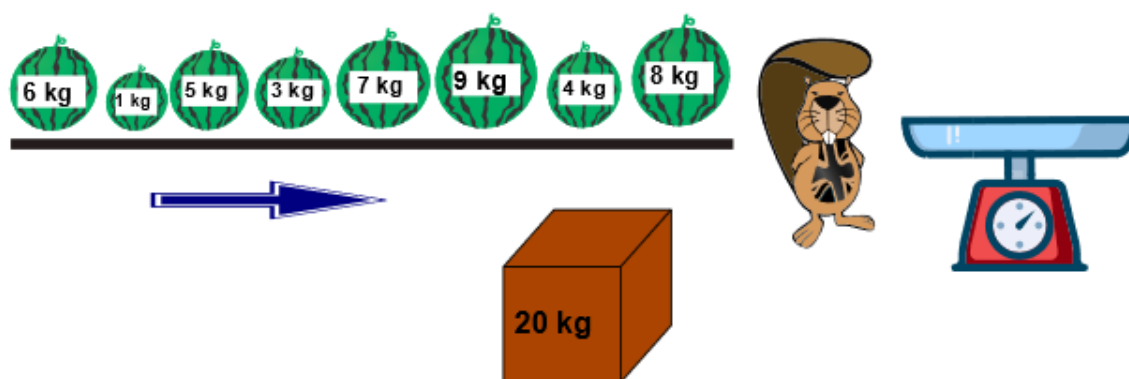


ZADATAK

Posao dabra Mirka je vaganje lubenica za isporuku kupcima. Mirko uzima jednu po jednu lubenicu s pokretne trake i stavlja je na vagu. Svaki put kad stavi novu lubenicu na vagu Mirko provjerava ukupnu težinu svih lubenica koje se nalaze na vagi.

Ako je ukupna težina manja ili jednaka 20 kg, lubenica ostaje na vagi.

Ako je ukupna težina veća od 20 kg, lubenica se uklanja s vage i ne stavlja u paket za isporuku. Kad ukupna težina svih lubenica koje se nalaze na vagi bude jednaka



74

PITANJE/IZAZOV

S obzirom na trenutno stanje na pokretnoj traci, koliko lubenica će biti isporučeno?

PONUĐENI ODGOVORI

- a) 3
- b) 4
- c) 5
- d) 6

TOČAN ODGOVOR

Točan odgovor je B) 4.

OBJAŠNJENJE

Prva lubenica ima težinu 8 kg i stavljena je na vagu. Stoga je ukupna težina na vagi 8 kg. Mirko zatim dodaje drugu lubenicu težine 4 kg. Ukupna težina na vagi sada iznosi 12 kg.

Dodavanjem treće lubenice težine 9 kg, ukupna težina na vagi postaje veća od 20 kg, zbog čega će treća lubenica biti uklonjena s vage.

Četvrta lubenica ima težinu 7 kg, zbog čega ukupna težina na vagi sada iznosi 19 kg.

Sljedeće dvije lubenice imaju težinu 3 kg i 5 kg i neće biti stavljene u paket jer bi u tom slučaju ukupna težina bila veća od 20 kg.

Sedma po redu lubenica ima težinu 1 kg. To će biti posljednja lubenica koja će biti stavljena na vagu jer će ukupna težina tada iznositi 20 kg.

Ukupno će biti isporučene 4 lubenice.

RAČUNALNA POVEZANOST

U ovom zadatku možemo uočiti dva ključna pojma iz područja programskih jezika i algoritama: pojam ponavljanja i provjere istinitosti uvjeta.

Uoči da se proces vaganja lubenica ponavlja sve dok se ne dostigne željena težina paketa za isporuku. To je primjer algoritma ponavljanja (koji se ponekad naziva iteracija).

Odluka koju dabar Mirko donosi za svaku lubenicu, a koja glasi: „Je li ukupna težina još uvijek manja od 20 kg?“ primjer je provjere istinitosti uvjeta. Ako je zadani uvjet zadovoljen, određeni događaj će se dogoditi. U ovom slučaju, događaj koji će se dogoditi je ostavljanje lubenice u paketu za isporuku.

While petlja sastoji se od bloka naredbi i uvjeta. Najprije se provjerava uvjet. Ako je uvjet zadovoljen (istinit) izvršavaju se naredbe unutar bloka naredbi. Postupak se ponavlja sve dok je uvjet zadovoljen (U ovom zadatku: kutija će se puniti sve dok je ukupna težina kutije manja od 20 kg).

U slučaju uvjetovanog izvršavanja naredbi (*if then*), ako je uvjet zadovoljen izvršavaju se naredbe koje se nalaze u *if* bloku (u ovom zadatku: nakon što je s pokretne trake uzeta nova lubenica, provjerava se je li ukupna težina paketa manja ili jednaka 20 kg. Ako jest, lubenica se dodaje u paket za isporuku).

```
a=0
k=0
while a<20:
    b=int(input("Koliko iznosi težina lubenice?"))
    if a+b<=20:
        a=a+b
        k=k+1
print(k)
```

```
int b,a=0,k=0;
while (a<20){
    cout<<"Koliko iznosi težina lubenice?";
    cin>>b;
    if ((a+b)<=20){
        a+=b;
        k++;
    }
}
cout<<k;
```

VIDEO KOMPRESIJA

Oznaka zadatka: 2019-RU-01

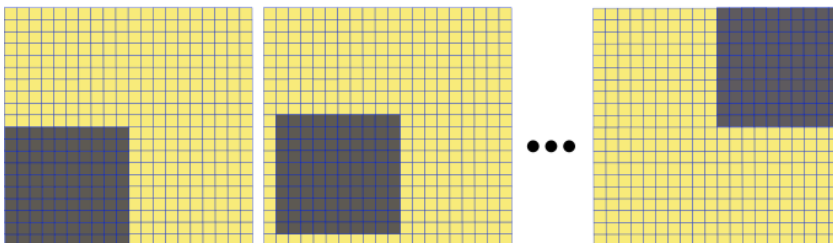
Tip pitanja: kratki odgovor (upis cijelog

Ključne riječi: kompresija



ZADATAK

Računalo pohranjuje sliku kao mrežu obojanih kvadratića koje zovemo pikseli. Video je niz slika (eng. frames), od kojih se svaka slika neznatno razlikuje od prethodne. Najjednostavniji



način pohrane videa je zapamtiti sve piksele iz svake slike. Bolji, efikasniji način je pohraniti samo čitavu prvu sliku, a zatim samo one piksele koji su različiti u dvije susjedne slike. U gornjoj slici, tamni kvadrat veličine 10x10 piksela kreće se iz donjeg lijevog kuta prema gornjem desnom kutu po kvadratu veličine 20x20 piksela, pomičući se po jedan piksel vodoravno i okomito u svakoj slici. Zato je potrebno 11 slika. Ako ovaj video spremamo u najjednostavnijem formatu, tad trebamo $(20 \times 20) \times 11 = 4400$ piksela.

PITANJE/IZAZOV

Ako ovaj video spremate u boljem, efikasnijem formatu, koliko piksela trebate za njegovu pohranu? Upišite samo broj piksela.

TOČAN ODGOVOR

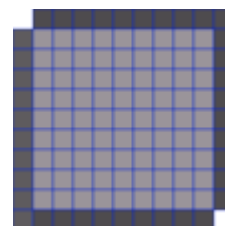
780

OBJAŠNENJE

Za pohranu prve slike treba $20 \times 20 = 400$ piksela.

U svakoj sljedećoj slici mijenja se 38 piksela:

Nakon prve slike slijedi 10 slika s po 38 piksela koje treba pohraniti pa je ukupan broj piksela $400 + 10 \times 38 = 780$ piksela.



RAČUNALNA POVEZANOST

Kompresija podataka je značajno područje informatike, posebno pri pohrani zvučnih i slikovnih zapisa. Neki od poznatih formata za pohranu fotografija, kao JPEG gube podatke pri kompresiji, pa neke boje ili rubovi budu izgubljeni. Stoga je zadatak napraviti algoritme kompresije kod kojih podaci o fotografiji neće biti izgubljeni.

Ako pohranjujemo sličice jednu za drugom, onda možemo svaku sličicu promatrati kao dio trodimenzionalnog polja. Na zaslonu se mijenjaju samo oni pikseli koji su različiti u dvije susjedne slike. Osim toga, u opisanom algoritmu pamti se samo položaj kvadratića, tj. broj piksela. Kad bi to željeli pretvoriti u jedinice za pohranu podataka (bitove, bajtove...) svaki piksel treba pomnožiti s odabranom dubinom boje, npr:

- jednoboja slika (crno bijela slika, 1 bit za svaki piksel): $780 \times 1 \text{ b} = 780 \text{ b}$
- bitmapa u 16 boja (4 bita za svaki piksel): $780 \times 4 \text{ b} = 3120 \text{ b} = 390 \text{ B}$
- bitmapa u 256 boja (8 bita za svaki piksel): $780 \times 8 \text{ b} = 6240 \text{ b} = 780 \text{ B}$
- bitmapa u 16 milijuna boja (24 bita = 3 B za svaki piksel): $780 \times 3 \text{ B} = 2340 \text{ B} \gg 2.29 \text{ kB}$

MOSTOVI I OTOCI

Oznaka zadatka: 2019-VN-05

Tip pitanja: Višestruki izbor

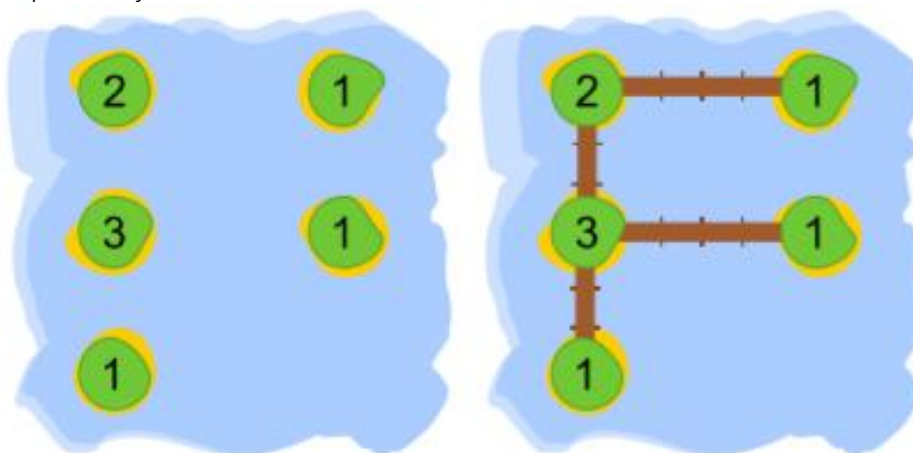
Ključne riječi: teorija grafova



ZADATAK

Krugovi na karti prikazuju otoke. Da bi mogao putovati između otoka, dabar mora sagraditi mostove. Brojevi navedeni na svakom otoku govore koliko je mostova povezanih s tim otokom. Mostovi se mogu graditi samo vodoravno ili okomito. Nakon što su svi mostovi izgrađeni, sustav mostova mora omogućiti putovanje s bilo kojeg otoka na bilo koji drugi otok.

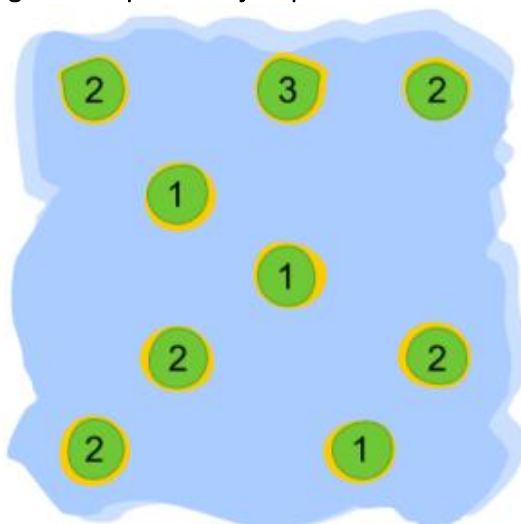
Na primjer, jedna skupina otoka je prikazana na karti lijevo. Sustav sa svim izgrađenim mostovima te skupine otoka prikazan je na karti desno.



77

PITANJE/IZAZOV

Koliko je mostova potrebno izgraditi na prikazanoj mapi?

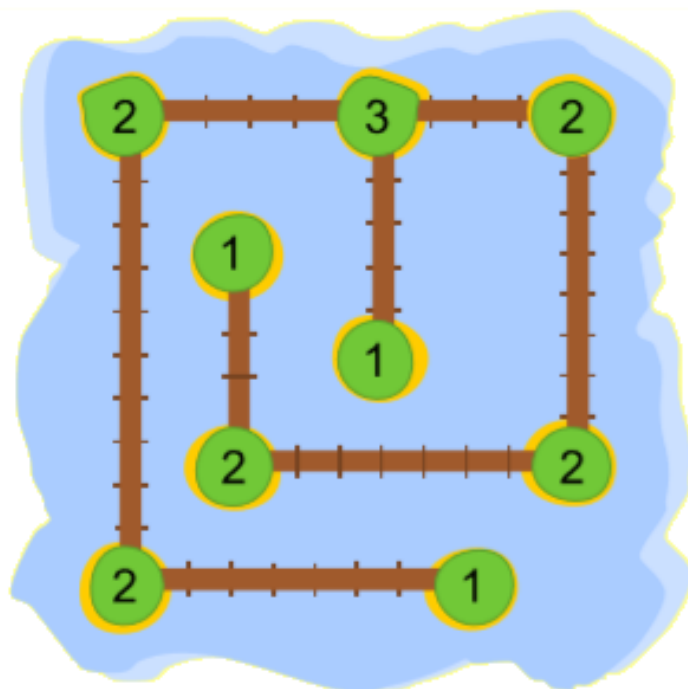


PONUĐENI ODGOVORI

- a) 7
- b) 8
- c) 9
- d) 10

TOČAN ODGOVOR

- b) 8



78

OBJAŠNENJE

U svakom dovršenom sustavu mostova, svaki se most računa na oba otoka koja povezuje. Dakle, da bi se dovršila zadana karta, broj mostova treba biti polovica ukupnog broja svih brojeva na karti, što je $(2 + 3 + 2 + 1 + 1 + 2 + 2 + 2 + 1) / 2 = 8$.

Za potpun odgovor, treba pokazati da je za datu kartu moguć takav cjelovit sustav mostova.

RAČUNALNA POVEZANOST

U teoriji grafova, jednoj od grana matematike, objekti predstavljeni točkama povezani su linijama kako bi označili parni odnos. Točke nazivamo vrhovi, a linije kojima su povezani nazivamo bridovi. Stupanj vrha označava broj bridova koji se spajaju u vrhu, slično broju mostova spojenih s otokom. Zbroj svih stupnjeva vrhova u grafu jednak je dvostrukom broju bridova, jer svaki brid ima dvije krajnje točke.

To se može proširiti i na "Lemu o rukovanju" koja kaže da svaki konačni neusmjereni graf ima parni broj vrhova s neparnim stupnjem (brojem bridova koji dodiruju vrh). Jednostavno rečeno, u grupi ljudi od kojih se neki rukuju, parni broj ljudi rukovao se neparnim brojem tuđih ruku.



Vidimo se i sljedeće godine, zar ne? 😊