

DABAR

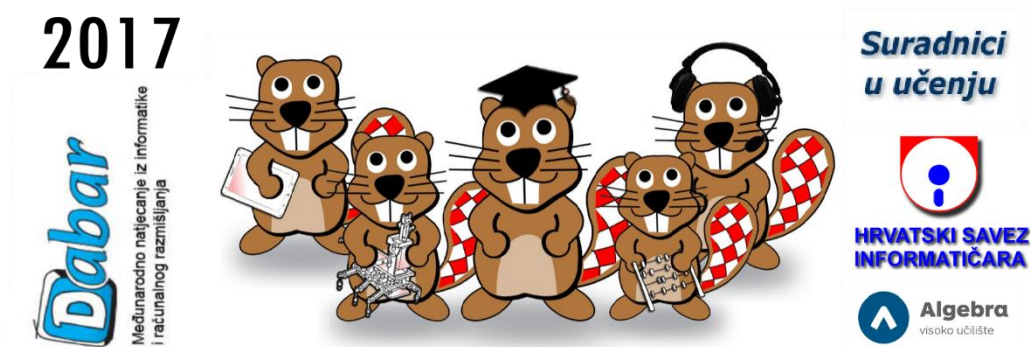
MEĐUNARODNO NATJECANJE IZ INFORMATIKE I
RAČUNALNOG RAZMIŠLJANJA



2017.
5017

Ucitelji.hr

Uvod



Drugu godinu za redom, Hrvatska je organizirala natjecanje iz informatike i računalnog razmišljanja – Dabar. Ovo međunarodno natjecanje podupire 58 zemalja iz cijeloga svijeta i ponosni smo što ove godine sudjelujemo kao punopravna članica Bebras zajednice.

Natjecanje je organizirano 13.-17.11.2017. na CARNet-ovom sustavu Loomen.

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

Organizator natjecanja za Hrvatsku je udruga “Suradnici u učenju” uz podršku Hrvatskog saveza informatičara i Visokog učilišta Algebra, CARNet-a te Ministarstva znanosti i obrazovanja.

Natjecanje se organizira u 5 kategorija:

- ✓ MikroDabar za učenike 1. i 2. razreda OŠ,
- ✓ MiliDabar za učenike 3. i 4. razreda OŠ,
- ✓ KiloDabar za učenike 5. i 6. razreda OŠ,
- ✓ MegaDabar za učenike 7. i 8. razreda i
- ✓ GigaDabar za učenike svih razreda srednje škole.

Neovisno o kategoriji, učenici su morali riješiti 15 zadataka uz vremensko ograničenje od 45 minuta.

Ove godine, Dabar je privukao čak 430 škola. Ukupno je sudjelovali 15247 učenika, što je čak 2,7 puta veći broj učenika nego 2016. godine. Broj učenika, po kategorijama je sljedeći:

- ✓ **MikroDabar** – 789 učenika 1. i 2. razreda OŠ
- ✓ **MiliDabar** – 2010 učenika 3. i 4. razreda OŠ
- ✓ **KiloDabar** – 4748 učenika 5. i 6. razreda OŠ
- ✓ **MegaDabar** – 4298 učenika 7. i 8. razreda OŠ
- ✓ **GigaDabar** – 3402 učenika svih razreda SŠ

U kategorijama KiloDabar i MegaDabar ostvaren je maksimalni broj bodova (15). Najveći broj bodova u kategoriji MikroDabar je 14.81, MiliDabar 13 i GigaDabar 14 bodova.

Za sve sudionike natjecanja pripremljene su diplome za sudjelovanje čiji je ispis i podjelu organizirala škola. 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.

I ove godine natjecanje je pripremao veliki broj volontera. Oni su čak 5 mjeseci prije samog natjecanja krenuli s radom kako bi natjecanje čim bolje pripremili za sve zainteresirane škole, učitelje/nastavnike i učenike. Njihov rad i entuzijazam je nemjerljiv jednim običnim hvala, no ipak HVALA VAM!

Veliko hvala i svim učiteljima i nastavnicima koji su se uključili u ovo natjecanje. Čak 80% učenika se ove godine priključilo po prvi puta i prema komentarima učenika i njihovih mentora, na dobrom smo putu da sljedeće godine budemo i bolji, te privučemo još veći broj natjecatelja.

Procjena težine zadataka prema ocjenama natjecatelja:

Procijeni težinu zadataka u svojoj kategoriji (1 - lagano, 5 - teško).						
						Average rank ↓
						1 2 3 4 5
Procjena za kategoriju označenu u 1. pitanju.						3.6
Odgovori	1	2	3	4	5	Ukupno
Procjena za kategoriju označenu u 1. pitanju.	277 (4%)	384 (6%)	2228 (36%)	2228 (36%)	1109 (18%)	6226

Koordinatorica natjecanja Dabar 2017:

Darija Dasović Rakijašić

Kontakt e-mail adresa: dabar@ucitelji.hr

Programski odbor natjecanja Dabar 2017:

Darija Dasović Rakijašić, Sanja Pavlović Šijanović, Vesna Tomić, Karolina Brleković, Sanda Šutalo i Marica Jurec.

Organizacijski odbor natjecanja Dabar 2017:

Danijel Forjan, Dubravka Petković, Aleksandra Žufić, Sanja Janeš, Loredana Zima Krnelić, Ela Veža, Renata Pintar, Nataša Ljubić Klemše, Darko Rakić, Gordana Sokol, Alma Šuto, Ivana Zlatarić, Sanela Jukić, Maja Kalebić, Nataša Glavor, Ljiljana Jeftimir, Kristina Lučić, Daniela Orlović, Sanja Hrvojević, Nikola Mihočka, Anica Leventić, Barbara Knežević, Daniela Usmiani, Daniel Rakijašić, Darka Sudarević, Iva Mihalić Krčmar, Valentina Pajdaković, Jelena Nakić, Dijana Kapus, Veronika Antal Horvat, Nataša Bek, Dominik Šutalo i Tamara Vučković.

U nastavku objavljujemo sve zadatke s natjecanja Dabar 2017. 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.

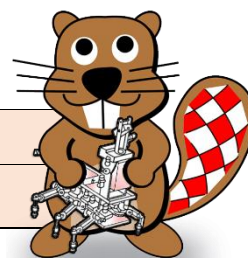


Sadržaj

GLAZBA.....	7
GRAD KRUŽNIH TOKOVA	8
UREĐIVANJE NOVINA	9
SUDOKU.....	10
KOJE JE TVOJE NINJA IME?	11
PEKARNICA	12
LOL TROL	13
KOLIKO PUTEVA?	14
KUĆICA ZA PTICE	15
KOLO SREĆE.....	16
DABROVA TRŽNICA.....	17
BLAGO	18
KRIŽANCI.....	19
LOV NA JAGODU.....	20
PET ŠTAPIĆA	21
INTEGRAM.....	23
ROĐENDANSKA PROSLAVA	26
DVIJE CESTE	28
DISTRIBUCIJA PADALINA	30
SAKUPLJANJE SLOVA	31
SKUP DABROVA	32
IDI KUĆI	35
KUĆICE ZA PTICE.....	36
BAKIN DŽEM	37
CRV	39
DABROVA BRANA	40
ZUBNA PUTOVNICA	42
PIGPEN ŠIFRA	43
SKRAĆIVANJE PORUKA	44
NAJSLAĐI ZALOGAJ	45
PAKIRANJE JABUKA	47
MREŽNA ABECEDA	48
RIBE	49
SPORTOVI	50
NATJECANJE DABROVA	51
SAKUPLJANJE SLOVA	53
CRVENO SVJETLO, PLAVO SVJETLO.....	54
LIJEVO - DESNO.....	56

ZAGRADE	57
BINARNI ULAZ.....	58
TVORNICA IGRAČKA	59
GRADNJA BRANE	61
IGRAČKE	62
PLES	64
SKRAĆIVANJE PORUKA	66
OPROŠTAJNA ZABAVA	67
DNEVNA MIGRACIJA.....	69
SAKUPLJANJE BOMBONA	70
IZLAZ IZ LABIRINTA	72
PERILICA RUBLJA	74
GRAFIKONI.....	76
PREDVIĐANJE POBJEDNIKA	77
UREDSKA RASVJETA.....	79
ŠARENA ZGRADA	80
S KAMENA NA KAMEN	82
RAČUNALNI CENTAR.....	83
SKAKUTAVE ŽABE	84
KUGLICE.....	85
CVIJEĆE ZA MARINU	87
LABIRINT.....	88
ANDREINA OGRLICA	90
SAVRŠENO MIJEŠANJE	92
DABROHOTEL	93
LEVENSHTEINOVA UDALJENOST	94
MONTAŽNA TRAKA	95
BRANA	96
SLAVNI DABAR.....	97
POVEZANE SOBE.....	98
HONOMAKATO ARHIPELAG.....	99
ZLATNA POLUGA	100
ATLETSKA EKIPA	102
RAZMJENA KNJIGA	103
NAJKRAĆA UDALJENOST.....	104
PROVALA	105
BISTRO	106

INTEGRAM



Oznaka zadatka: 2017-HR-03

Tip pitanja: interaktivno

Ključne riječi: Booleova algebra, logika

ZADATAK

Naši su dabrovi svestrani te uče programiranje i strani jezik istovremeno.

Svaki dabar zna jedan strani i jedan programski jezik.

- dabar Luka zna hrvatski jezik, ali, ne zna Python,
- dabar Darko zna njemački jezik, ali ne zna C++,
- David ne zna talijanski jezik i ne zna ni Javu,
- Davor ne zna francuski, a niti ne programira u C++,
- dabar koji programira u Logu, govori talijanski,
- dabar koji govori francuski ne zna programirati u C++.

PITANJE/IZAZOV

Ako znamo da svaki dabar govori jedan strani jezik i programira u jednom od navedenih programskih jezika popunite tablicu:

	Python	C++	Java	Logo
Hrvatski				
Njemački				
Francuski				
Talijanski				

Luka	Darko	David	Davor
------	-------	-------	-------

TOČAN ODGOVOR

	Python	C++	Java	Logo
Hrvatski		Luka		
Njemački			Darko	
Francuski	David			
Talijanski				Davor

- Luka govori hrvatski i programira u C++.
- Darko poznaje njemački jezik i programira u Javi.
- David zna francuski jezik i Python.
- Davor je naučio talijanski i Logo.

OBJAŠNJENJE

Koraci:

	Hrvatski	Njemački	Francuski	Talijanski
Luka	+	-	-	-
Darko	-	+	-	-
David	-	-	+	-
Davor	-	-	-	+

	Python	C++	Java	Logo
Luka	-	+	-	-
Darko	-	-	+	-
David	+	-	-	-
Davor	-	-	-	+

	Python	C++	Java	Logo
Hrvatski		Luka		
Njemački			Darko	
Francuski	David			
Talijanski				Davor

Može i pomoću Integram tablice:

		Programski jezik				Strani jezik			
		Python	C++	Java	Logo	Hrvatski	Njemački	Francuski	Talijanski
DABAR	Luka								
	Darko								
	David								
	Davor								
Strani jezik	Hrvatski								
	Njemački								
	Francuski								
	Talijanski								

Popunimo početnu tablicu s onim što nam je poznato

		Programski jezik				Strani jezik			
		Python	C++	Java	Logo	Hrvatski	Njemački	Francuski	Talijanski
DABAR	Luka	-				+	-	-	-
	Darko		-			-	+	-	-
	David			-		-	-	+	-
	Davor		-			-	-	-	+
Strani jezik	Hrvatski								
	Njemački								
	Francuski		-						
	Talijanski				+				

Sad možemo zaključiti :

- Davidu je ostao francuski jezik, a Davoru talijanski te smo popunili tablicu Dabar/Jezik.
- Kako Davor govori talijanski, a dabar koji zna talijanski programira u Logu zaključujemo da Davor programira u Logu.
- David zna francuski, ali Dabar koji zna francuski ne programira u C++ , pa David ne programira u C++. Ostaje nam jedino Luka koji programira u C++.

		Programski jezik				Strani jezik			
		Python	C++	Java	Logo	Hrvatski	Njemački	Francuski	Talijanski
DABAR	Luka	-	+	-	-	+	-	-	-
	Darko	-	-	-	-	-	+	-	-
	David	-	-	-	-	-	-	+	-
	Davor	-	-	-	+	-	-	-	+
Strani jezik	Hrvatski								
	Njemački								
	Francuski		-						
	Talijanski				+				

David programira u Pythonu, a Darko u Javi.

		Programski jezik				Strani jezik			
		Python	C++	Java	Logo	Hrvatski	Njemački	Francuski	Talijanski
DABAR	Luka	-	+	-	-	+	-	-	-
	Darko	-	-	+	-	-	+	-	-
	David	+	-	-	-	-	-	+	-
	Davor	-	-	-	+	-	-	-	+
Strani jezik	Hrvatski								
	Njemački								
	Francuski		-						
	Talijanski				+				

Nadalje tablicu dovršimo na način da u svakoj manjoj tablici imamo samo jedan plus u svakom retku i stupcu.

		Programski jezik				Strani jezik			
		Python	C++	Java	Logo	Hrvatski	Njemački	Francuski	Talijanski
DABAR	Luka	-	+	-	-	+	-	-	-
	Darko	-	-	+	-	-	+	-	-
	David	+	-	-	-	-	-	+	-
	Davor	-	-	-	+	-	-	-	+
Strani jezik	Hrvatski	-	+	-	-				
	Njemački	-	-	+	-				
	Francuski	+	-	-	-				
	Talijanski	-	-	-	+				

RAČUNALNA POVEZANOST

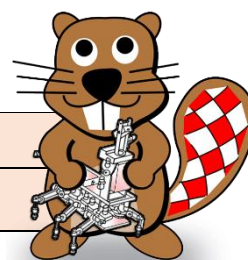
Matematička logika proučava osnovne principe matematičkog zaključivanja. Booleova je algebra dio matematičke logike. Logička ili Booleova algebra je sustav teorema koji rabe simboličku logiku kako bi opisali skupove elemenata i odnose među njima. Temelji se na principu deduktivnog logičkog zaključivanja, ulazni podaci mogu imati samo dva stanja: točno i netočno. Booleova algebra primjenjiva je u konstruiranju i analizi rada računala jer računala također imaju samo dva stanja: uključen – isključen, maksimalan napon – minimalan napon... Osnovni element logičke algebre je IZJAVA koja može biti istinita (0) ili lažna (1). Tablica stanja izražava odnose između operanda ovisno o logičkoj operaciji. Tablica stanja definicija je logičke operacije i mora sadržavati sva moguća stanja operanada i logičke operacije.

ROĐENDANSKA PROSLAVA

Oznaka zadatka: 2017-SK-07

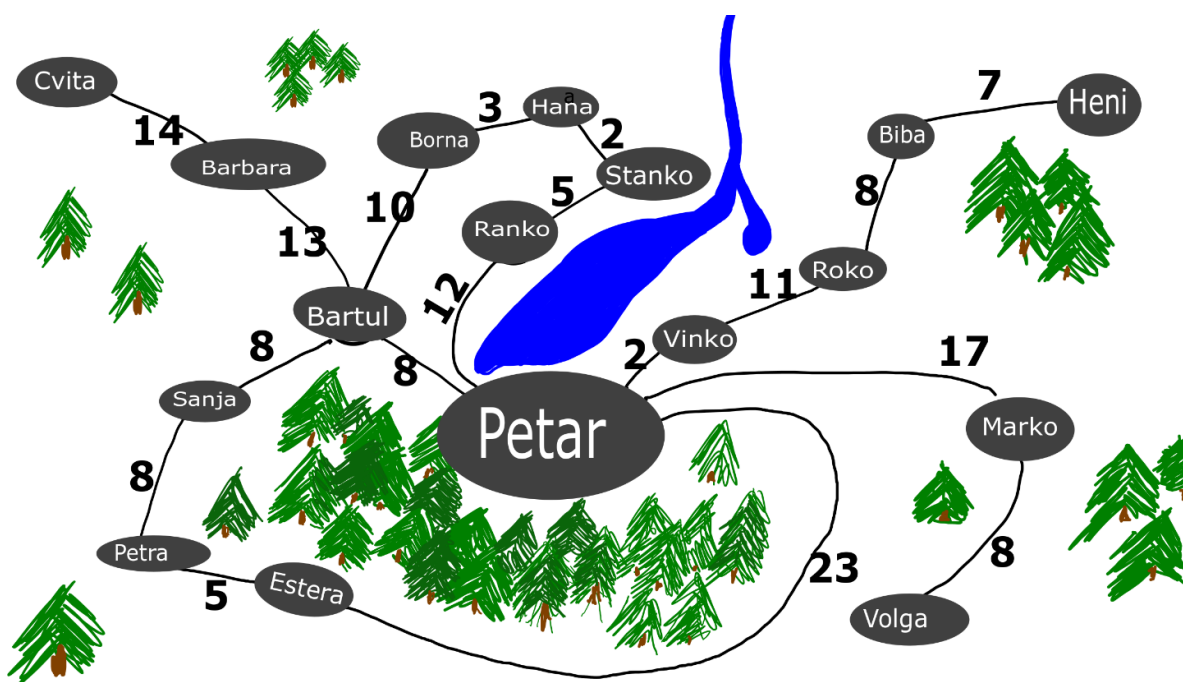
Tip pitanja: interaktivno

Ključne riječi: struktura podataka, putanja, pretraživanje grafa



ZADATAK

Petar živi u Dabrogradu i pozvao je prijatelje iz susjednih mjesta na rođendansku proslavu. Međutim, na proslavu su pristigli samo prijatelji koji žive na udaljenosti do 20 km od Petra.



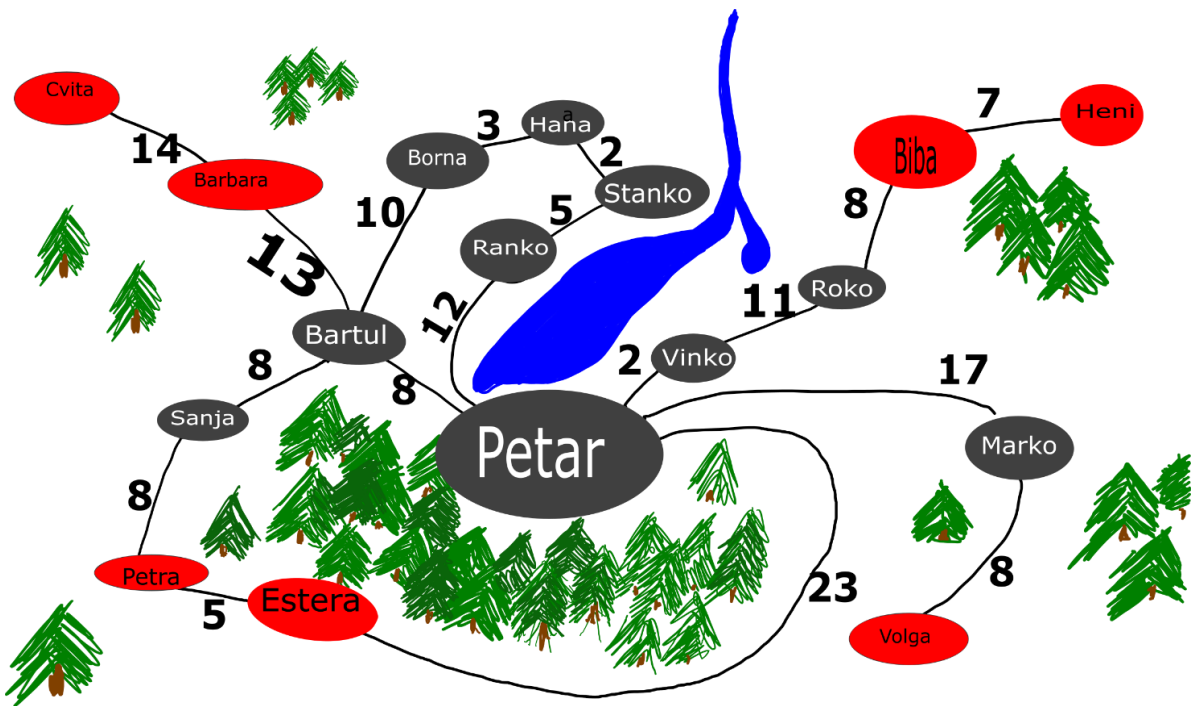
Na crtežu su prikazane udaljenosti između mjesta.

PITANJE/IZAZOV

Potrebno je popuniti sva mjesta na slici i to na sljedeći način: imena u crvenim i sivim kvadratićima postavite na odgovarajuća mjesta, s tim da su crvenom bojom označeni oni koji nisu stigli na proslavu a sivom bojom oni koji su stigli jer žive na udaljenosti do 20 km od Petra.

TOČAN ODGOVOR

Točni odgovori označeni su na slici. Crvenom bojom označeni su Petrovi prijatelji koji nisu stigli na proslavu, dok su sivom bojom označeni svi Petrovi prijatelji koji su na proslavu stigli.



OBJAŠNENJE

Put od Dabrograda do svakog susjednog mjesta može se izračunati pronalaženjem puta koji ih povezuje, pri čemu je potrebno zbrojiti njihove udaljenosti.

Kako bismo bili sigurni da ćemo sustavno provjeriti sve udaljenosti, koristimo redoslijed provjeravanja od najbližeg mjesta ka najudaljenijem. Za rješavanje problema pronalaženja najkraćeg puta često se koristi tzv. Dijkstrin algoritam. Evo, kako njegova primjena izgleda u našem zadatku:

Vinkova udaljenost iznosi 2 km.

Bartul je udaljen 8 km.

Ranko je udaljen 12 km.

Rokova udaljenost iznosi 13 km = 2km + 11km, a Biba je predaleko 13km + 8km = 21km.

Sanja je stigla puteljkom duljine 16 km = 8km + 8km, dok je Petra predaleko 16km + 8km = 24km.

Marko je udaljen 17km, ali je Volga predaleko 17km + 8km = 25km.

Stanko je stigao preko Ranka 17km = 12km + 5km.

Borna je udaljen 18 km = 8km + 10km, dok je Hana putanjom 18 km + 3km = 21 km, predaleko, ali je drugom putanjom 17km + 2km udaljena samo 19km.

Video dodatak- Dijkstrin algoritam: <https://www.youtube.com/watch?v=aBym6p-coic>.

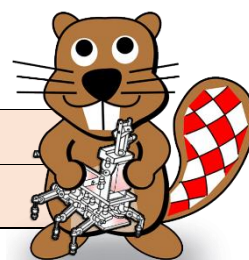
RAČUNALNA POVEZANOST

Informatika je znanost koja se bavi i pohranjivanjem i prikazivanjem informacija. U ovom zadatku koristimo kartu koja predstavlja distribuciju udaljenosti pojedinih mjesta. Koristeći matematička znanja možemo izračunati i usporediti udaljenosti pojedinih mjesta dok znanjima iz računalne znanosti, možemo pronaći susjedna mjesta koja ispunjavaju unaprijed zadane uvjete.

Detaljnije objašnjenje može se pronaći na

<https://www.khanacademy.org/computing/computer-science/algorithms/breadth-first-search/a/breadth-first-search-and-its-uses>

DVIJE CESTE



Oznaka zadatka: 2017-HR-01

Tip pitanja: interaktivno

Ključne riječi: logika, Booleova algebra

ZADATAK

Dabrograd i Drvograd povezani su dvjema cestama. Ceste imaju ograničenja za određena vozila. Vozeći prvom cestom prelazi se preko mosta. Druga cesta prolazi kroz tunel i uključuje i vožnju trajektom. Prvom cestom mogu prometovati vozila koja prevoze do 7 tona tereta. Druga cesta ima čak dva ograničenja.

Kroz tunel mogu proći vozila do 300 cm visine, a trajektom se mogu prevoziti samo vozila čija duljina ne prelazi 7 m.

Svako vozilo na sebi ima oznaku visine u centimetrima, težine u tonama i duljine u metrima. Na primjer vozilo na slici je 180 cm visoko, 4 tone teško i 6 metara dugo.



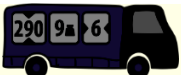
PITANJE/IZAZOV

Odaberi za svako vozilo koju cestu može koristiti. Neka vozila ne mogu proći niti jednom cestom, a neka mogu koristiti obje.

TOČAN ODGOVOR

Točni odgovori su:

	Vozilo	Cesta 1 (most)	Cesta 2 (tunel i trajekt)
A (180, 4t, 6m)		☒	☒
B (200, 8t, 9m)		☐	☐
C (190, 3t, 4m)		☒	☒

D (205, 5t, 5m)		☒	☒
E (210, 8t, 6m)		☐	☒
F (210, 2t, 8m)		☒	☐
G (290, 9t, 6m)		☐	☒
H (310, 5t, 5m)		☒	☐

OBJAŠNJENJE

Obje ceste mogu koristiti vozila čija je visina $\leq 3\text{m}$, težina $\leq 7\text{t}$ i duljina $\leq 6\text{m}$.

Prvom cestom prometuju vozila koja teže $\leq 7\text{t}$ (visina $> 3\text{m}$ i duljina $> 6\text{m}$).

Drugom cestom prometuju vozila koja teže $> 7\text{t}$, ali visine $\leq 3\text{m}$ i duljine $\leq 6\text{m}$.

Vozila koja ne mogu stići iz jednog mjesta u drugo su ona čija je visina $> 3\text{m}$, težina $> 7\text{t}$ i duljina $> 6\text{m}$.

Ako su svi uvjeti zadovoljeni (visina $< 300\text{cm}$, masa $< 7\text{t}$ i duljina $< 6\text{m}$) vozilo može koristiti obje ceste. To su vozila A, C i D.

Vozilo B ne zadovoljava niti jedan od uvjeta te ne može prometovati na ovoj relaciji.

Vozilo E može koristiti samo drugu cestu jer je preteško za most.

Vozilo F može koristiti samo prvu cestu jer je predugo za trajekt.

Vozilo G je preteško za most na prvoj cesti te mu ostaje druga cesta.

Vozilo H je previsoko za tunel, ali može prijeći most i koristiti prvu cestu.

RAČUNALNA POVEZANOST

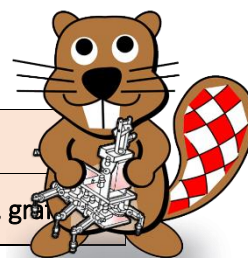
Booleova algebra važna je za računalnu znanost jer se podudara s binarnim brojevnim sustavom, pri čemu svaka izjava i svaki bit imaju vrijednost 1 ili 0 (odnosno istina (TRUE) ili laž (FALSE)).

Ovakvi zadaci uvod su u Booleove izraze koji se koriste u programskim jezicima, a svaki izraz na kraju daje Booleovu vrijednost i to točno jednu vrijednost istinu (1) ili laž (0).

Booleov izraz može se sastojati od kombinacije logičkih varijabli, konstanti istinitih ili lažnih, vrijednosti operatora i logičkih vrijednosti funkcija.

U ovom zadatku svaka cesta može se prikazati koristeći Booleove izraze, npr. za cestu 1: (vozilo visine $\leq 2\text{m}$) I (vozilo dužine $\leq 7\text{m}$).

DISTRIBUCIJA PADALINA



Oznaka zadatka: 2017-HU-04

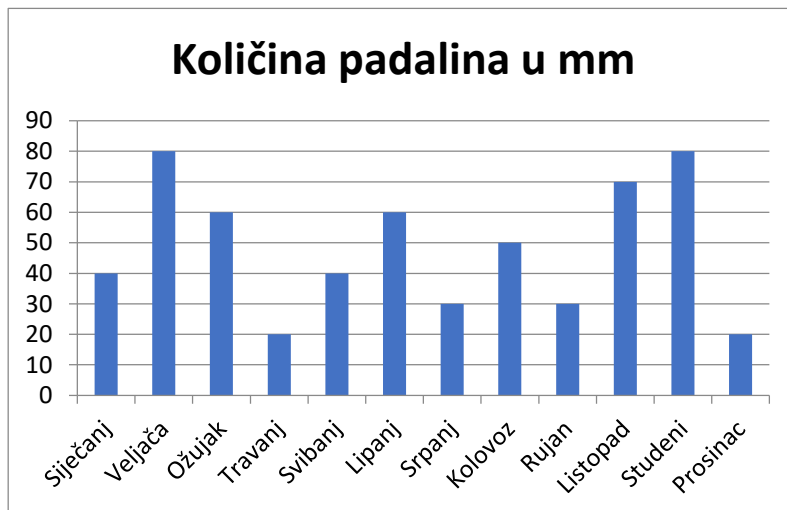
Tip pitanja: višestruki izbor (tekst)

Ključne riječi: najveća vrijednost, analiza podataka, dvostruko sortiranje, dijagram, grafi

ZADATAK

Dabrovi planiraju izgraditi novu branu. Na temelju prosječne godišnje količine padalina (donja slika) žele pronaći najpovoljniji mjesec za njegovu izgradnju. Potrebno je zadovoljiti sljedeće uvjete:

- Maksimalni kapacitet brane potrebno je testirati u najkišovitijem mjesecu cijele godine.
- Branu bi trebalo izgraditi mjesec ili dva prije testiranja i to u najsušem mogućem mjesecu.



PITANJE/IZAZOV

Koji mjesec je najpovoljniji za izgradnju nove brane?

PONUĐENI ODGOVORI:

- a) siječanj
- b) travanj
- c) rujan
- d) prosinac

TOČAN ODGOVOR

Prosinac

OBJAŠNJENJE

Točan odgovor je prosinac. Najprije ćemo potražiti najkišovitije mjesece. To su veljača i studeni. Znamo da bi se brana trebala graditi mjesec ili dva prije jednog od tih mjeseci, tj. u prosincu, siječnju, rujnu ili listopadu. Od navedenih bi trebalo odabrati najsušiji, a to je prosinac.

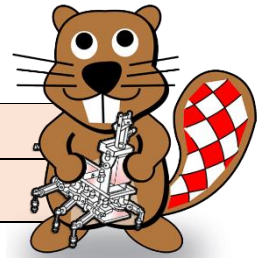
RAČUNALNA POVEZANOST

Analiza podataka kao i traženje odgovarajućeg (optimalnog) rješenja zajednička je aktivnost svih znanstvenika. Pri tome, važno je uzeti u obzir niz postavljenih uvjeta i to prema zadanom redoslijedu.

(Diagram wiki: <https://en.wikipedia.org/wiki/Diagram>)

Bar chart wiki: https://en.wikipedia.org/wiki/Bar_chart)

SAKUPLJANJE SLOVA



Oznaka zadatka: 2017-CY-04

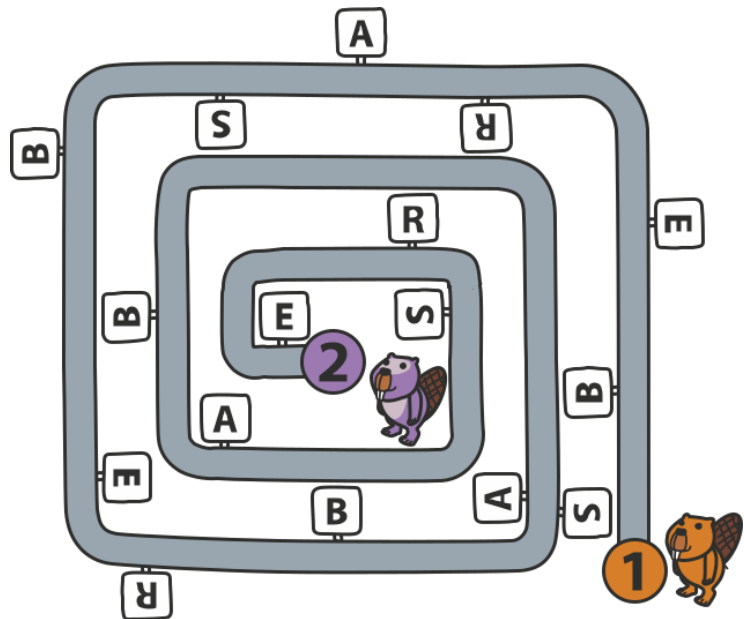
Tip pitanja: višestruki odgovor

Ključne riječi: string, znakovi, niz

ZADATAK

Zoki i Piko žele proći stazom kao što je prikazano na slici. Zoki želi ići od točke 1 do točke 2, a Piko želi ići od točke 2 do točke 1.

Uz stazu su postavljena slova (B, E, R, A i S) koja Zoki i Piko moraju sakupiti. Zoki i Piko mogu čitati samo slova koja se nalaze s njihove lijeve strane. Kad Zoki i Piko dođu do kraja puta imat će sakupljene nizove slova.



PITANJE/IZAZOV

Koje nizove slova će dabrovi sakupiti na kraju puta?

PONUĐENI ODGOVORI

- | | |
|---------------------|----------------|
| a) Zoki: BRSBAASE, | Piko: RBSRBAE |
| b) Zoki: BRSEBAASE, | Piko: RBBSRBAE |
| c) Zoki: BRSEBAASE, | Piko: RBSRBAE |
| d) Zoki: BRSEBAAS, | Piko: RBSRBE |

TOČAN ODGOVOR

Točan odgovor je C.

Zoki će sakupiti niz slova BRSEBAASE, a Piko će sakupiti niz slova RBSRBAE.

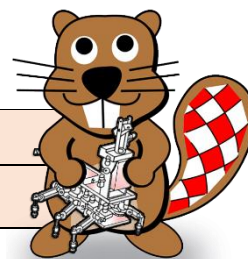
OBJAŠNJENJE

Zokiju su s lijeve strane redom slova BRSEBAASE, a Piko su s lijeve strane redom slova RBSRBAE.

RAČUNALNA POVEZANOST

Stringovi se često koriste u programiranju. String je vrsta podatka u obliku niza znakova (simbola, slova, brojk) promjenjive duljine. String može označavati i općenitije nizove ili popise. U programskim jezicima string je konačan niz znakova koji su izabrani iz skupa kojega zovemo abeceda.

SKUP DABROVA



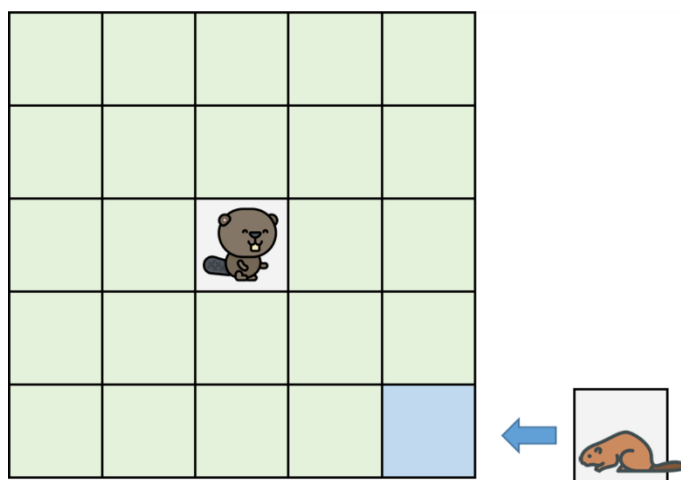
Oznaka zadatka: 2017-HU-08

Tip pitanja: višestruki odgovor

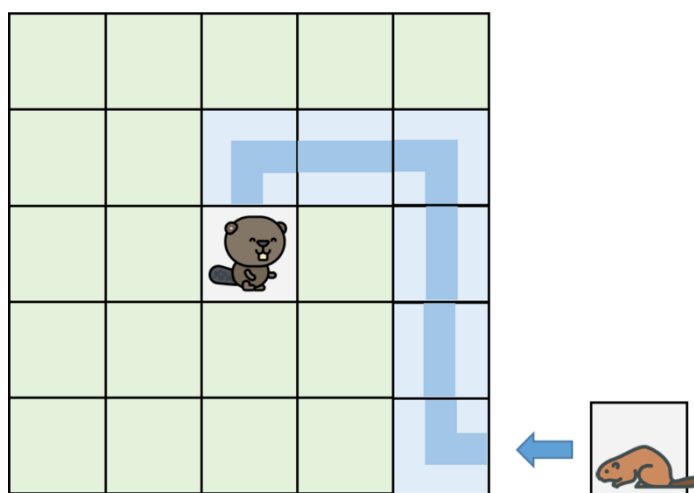
Ključne riječi: odrediti put, najbolji put, niz

ZADATAK

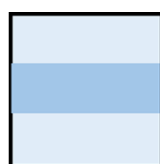
Pomozite mami dabrici da dođe do svojih dabrića. Sgradite put od ponuđenih karata.



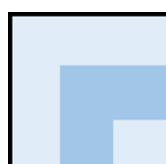
Na slici je jedan od mogućih putova:



Postoje samo dvije vrste karata:



Karta A



Karta B

Put se u primjeru može izgraditi sljedećim nizom karata: BAABAB. Karte se smiju okrenuti, ali ne smijemo mijenjati njihov poredak.

PITANJE/IZAZOV

Koji niz karata **neće** dovesti mamu dabricu do dabrića?

PONUĐENI ODGOVORI

- a) BBAABB
- b) ABBABB
- c) BBAAAA
- d) AAABAB

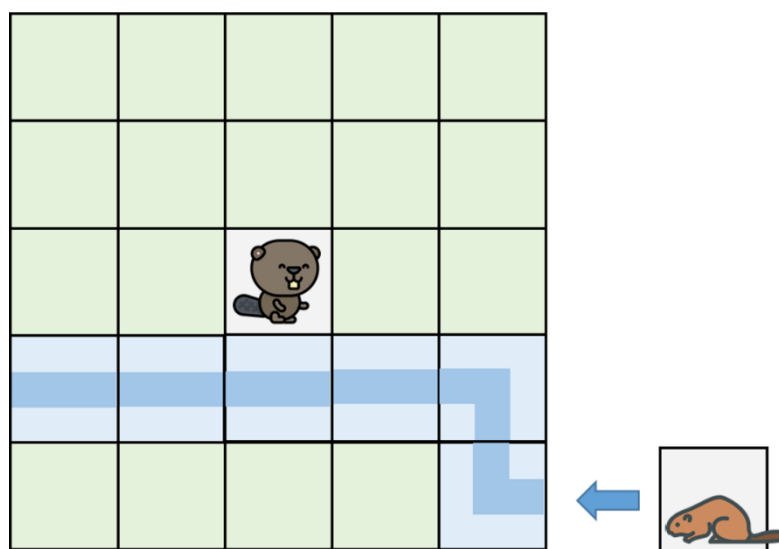
TOČAN ODGOVOR

- c) BBAAAA

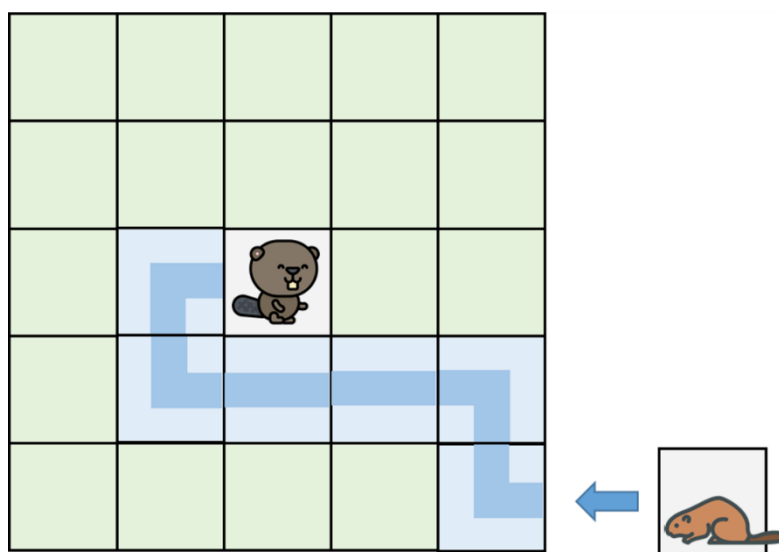
OBJAŠNJENJE

Ako je niz karata BBAAAA, tada je jedini mogući put kao na slici:

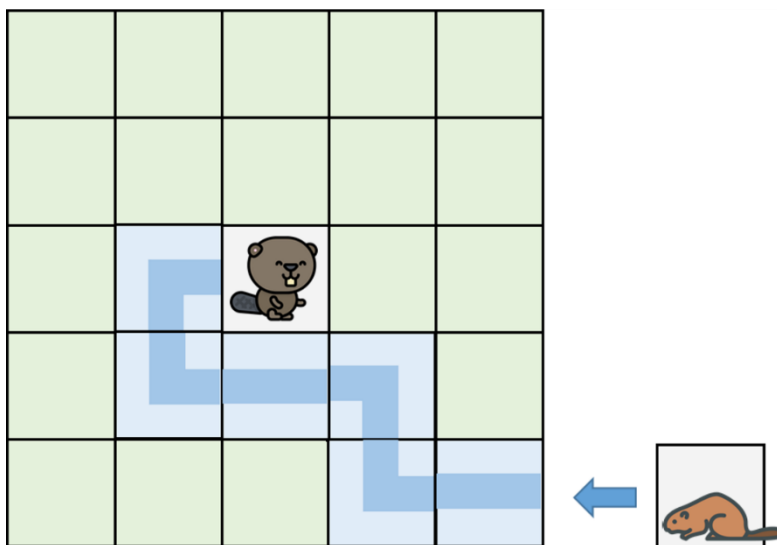
U ostala tri slučaja, postoje mogući putovi do malih dabrova.



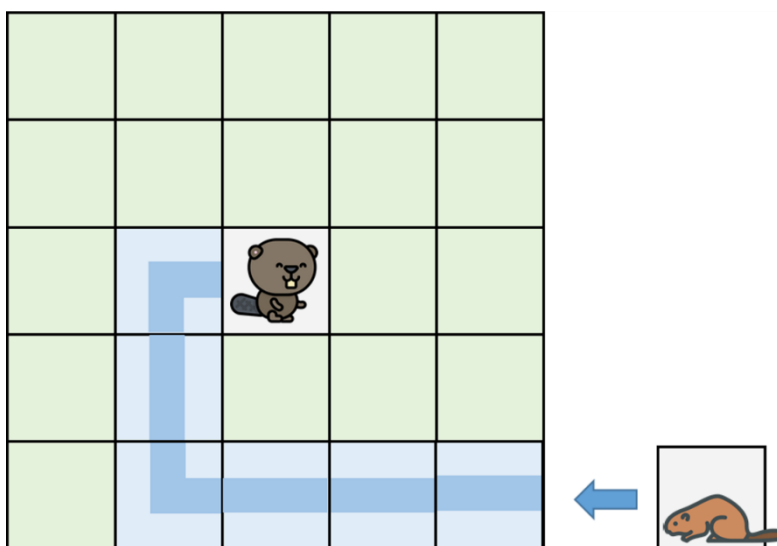
Odgovor a)



Odgovor b)



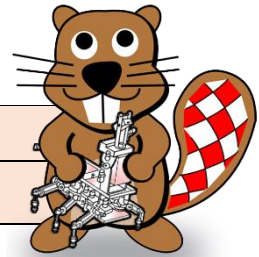
Odgovor d)



RAČUNALNA POVEZANOST

Ovaj zadatak koristi praćenje niza koraka. Nizovima se kod programiranja koristimo kada radimo s većim skupinama sličnih podataka, npr. imena učenika istog razreda, ocjene po predmetima za pojedinog učenika i sl. U navedenim slučajevima, kada se ne bismo koristili nizovima, morali bismo u programu rabiti mnogo varijabli različitih naziva.

IDI KUĆI



Oznaka zadatka: 2017-PK-03

Tip pitanja: višestruki odgovor

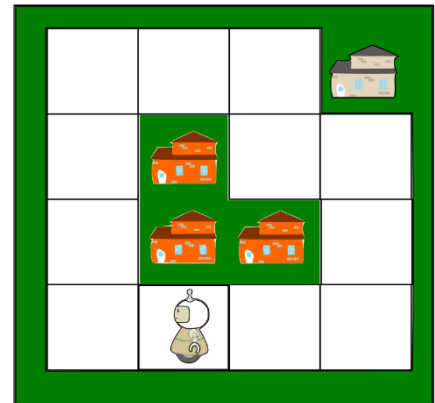
Ključne riječi: robot, program

ZADATAK

Policajac robot radi u Dabrogradu i svake noći obilazi ulice kružnim putem (pogledaj kartu) pri čemu koristi samo tri vrste kretanja: NAPRIJED, LIJEVO (bez kretanja naprijed), DESNO (bez kretanja naprijed).

PITANJE/IZAZOV

Koji je najmanji broj naredbi potreban da bi policajac robot obišao sve ulice i vratio se na mjesto s kojega je krenuo?



PONUĐENI ODGOVORI

- a) 11
- b) 16
- c) 12
- d) 18

TOČAN ODGOVOR

Najmanji broj naredbi potreban da bi policajac robot obišao sve ulice i vratio se na mjesto s kojega je krenuo iznosi 18.

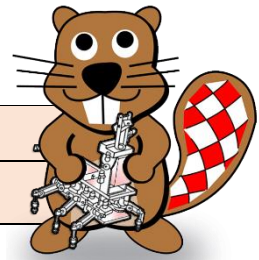
OBJAŠNJENJE

Samo naredba NAPRIJED pomiče policajca robota. Za promjenu smjera potrebne su dvije naredbe, jedna za okretanje i jedna za pomicanje robota naprijed. Staza ima 12 polja i 6 okretanja, što znači da je potrebno ukupno 18 naredbi da bi policajac robot prošao puni krug. Na primjer, ako naredbe označimo slovima (N-NAPRIJED, L-LIJEVO, D-DESNO), policajac robot će obići sve ulice i vratiti se na početno mjesto sljedećim nizom naredbi: NDNNDNNDNLNDNNDNN.

RAČUNALNA POVEZANOST

Policajac robot je računalni i programibilni sustav. Za programiranje policajca robota, moguće su samo tri naredbe. To izgleda vrlo jednostavno no pritom treba voditi računa o zadanim uvjetima. U informatici želimo proizvoditi sustave u kojima ćemo do rješenja doći s najmanjim brojem naredbi jer to smanjuje troškove sustava. Na taj način, sustav će izvršiti sve potrebno, a provedba naredbi neće biti nepotrebno skupa.

KUĆICE ZA PTICE



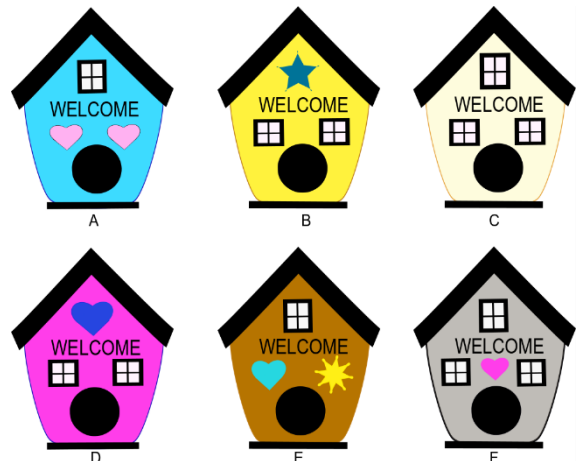
Oznaka zadatka: 2017-RO-03b

Tip pitanja: višestruki izbor (slika)

Ključne riječi: uzorak, vjerojatnost, apstrakcija

ZADATAK

Mama Darija želi kupiti kućicu za ptice kao poklon svojoj kćerki Loredani za rođendan. Na mamin upit kakvu točno kućicu za ptice želi, Loredana je odgovorila: „Željela bih kućicu za ptice s dva prozora i srcem.“ Saznavši Loredanine želje, mama Darija otišla je u trgovinu i odabrala savršenu kućicu za ptice.



PITANJE/IZAZOV

Koju od kućica je mama Darija kupila svojoj kćeri?

TOČAN ODGOVOR

Točan odgovor je kućica D koja ima 2 prozora i srce.

OBJAŠNJENJE

A nije točan odgovor jer ima kućica ima samo jedan prozor i dva srca.

B nije točan odgovor iako kućica ima 2 prozora, ali nema srce.

C nije točan odgovor jer kućica ima 3 prozora i nema srce.

E nije točan odgovor jer ima samo jedan prozor.

F nije točan odgovor jer kućica ima 3 prozora.



RAČUNALNA POVEZANOST

Povežimo zadatak s osnovnim informatičkim pojmovima ovog zadatka: specifičan problem/ algoritam, specifična struktura podataka i temeljna ideja. Važnost ovih pojmova izražena je u stvaranju računalnih programa, radu digitalnih sustava i u istraživanjima u području računalne znanosti.

Za rješavanje ovog zadatka, koristili smo jedan od temeljnih misaonih procesa a to je apstrakcija.

„U znanosti, a posebno u matematici, apstrakcija označuje djelotvoran i logički razrađen postupak za teoretsko upoznavanje predmeta i praktičko ovladavanje njime.“ (Iz članka prof. Kurnika na <http://mis.element.hr/fajli/113/06-04.pdf>)

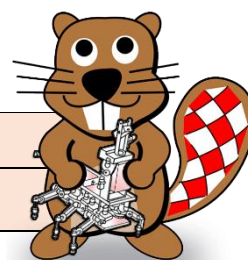
Apstrakcija uključuje pronalaženje (filtriranje) nepotrebnih detalja i izbor podataka koji su potrebni za rješavanje problema. U ovom postupku, može nam pomoći isticanje sličnosti i razlika.

Ovaj zadatak temelji se na prepoznavanju objekata s određenim detaljima (broj prozora, srce, zvjezdica), te se na osnovu tih detalja, objekti postepeno isključuju. Na kraju ostaje objekt koji će imati tražene detalje odnosno svojstva. U našem zadatku, to je kućica koja ima 2 prozora i srce.

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

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

BAKIN DŽEM



Oznaka zadatka: 2017-RU-04

Tip pitanja: višestruki izbor (slikom)

Ključne riječi: plan rada, pohlepni algoritam

ZADATAK

Ana, Petar i Liza pomažu baki napraviti džem. Za jednu staklenku džema moraju uraditi tri stvari:

- oprati staklenku, potrebno vrijeme 3 minute
- napuniti staklenku džemom, potrebno vrijeme 2 minute
- zatvoriti staklenku, potrebno vrijeme 3 minute

Moraju biti pažljivi: staklenka mora biti čista da bi se u nju mogao uliti džem i mogu zatvoriti samo staklenke koje su napunjene.

To znači da nije moguća situacija kao u primjeru na slici ispod:

ANA					
PETAR					
LIZA					

PITANJE/IZAZOV

Izaberite plan rada od 10 minuta kojim će se napuniti i zatvoriti najveći mogući broj staklenki.

a)

ANA					
PETAR					
LIZA					

b)

ANA					
PETAR					
LIZA					

c)

ANA				
PETAR				
LIZA				

TOČAN ODGOVOR

Jedan od točnih odgovora je:

ANA					
PETAR					
LIZA					

OBJAŠNJENJE

Primijetimo da je za jednu napunjenu i zatvorenu staklenku potrebno $3+2+1=6$ minuta. Za svakoga od tri pomagača imamo po 10 minuta, što je ukupno 30 minuta rada i prema tome nije moguće napuniti i zatvoriti više od $30:6=5$ staklenki.

Razmotrimo sada kako sve možemo oprati, napuniti i zatvoriti 5 staklenki. Treba naglasiti da se ne mogu puniti staklenke dok nisu oprane i da se ne mogu zatvoriti staklenke koje nisu napunjene.

Neka svatko od bakinih pomagača radi sljedeće:

- pere staklenke sve dok nema opranih za punjenje,
- puni staklenke sve dok nema napunjenih,
- zatvara napunjene staklenke.

Ovaj nas *pohlepni algoritam* dovodi do rješenja koje je prikazano na gornjoj slici.

Primijetimo da u zadatku nije važno koliko staklenki treba napuniti, ako bakini pomagači rade svatko za sebe, tada će stići napuniti samo tri staklenke.

ANA					
PETAR					
LIZA					

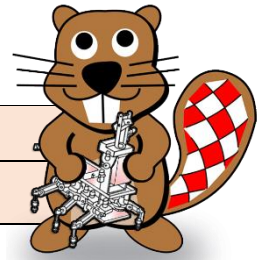
Moguće su i druge pogreške. Ako su bakini pomagači specijalizirani za određeni posao (npr. Ana samo pere staklenke, Petar ih samo puni, a Liza samo zatvara), tada Liza ne može zatvoriti staklenke jer još nisu napunjene. U tom slučaju bit će napunjene i zatvorene samo tri staklenke.

ANA					
PETAR					
LIZA					

RAČUNALNA POVEZANOST

Ovaj je zadatak povezan s izradom mrežnog plana i istodobnom obradom nekoliko operacija. Ispravnim radnim planom postiže se maksimalni učinak. Za rješenje problema koristili smo *pohlepni algoritam*.

CRV



Oznaka zadatka: 2017-SK-08

Tip pitanja: višestruki odgovor

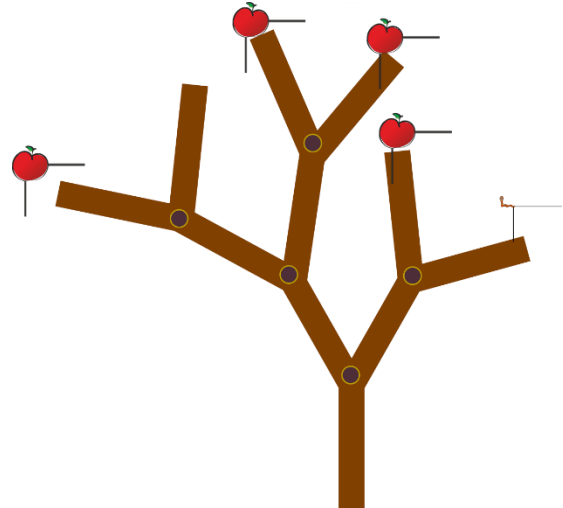
Ključne riječi: najkraći put, graf, stablo, optimizacija

ZADATAK

Na jednoj grani velikog stabla- sjedi crv. Može se kretati (penjati i spuštati) po svim granama stabla da bi pojeo sve jabuke. Svaki dio grane je duljine jedan metar.

PITANJE/IZAZOV

Koliko najmanje dijelova grana crv mora prijeći ako želi pojesti sve jabuke?



PONUĐENI ODGOVORI

- a) 4
- b) 6
- c) 13
- d) 15

TOČAN ODGOVOR

- c) 13

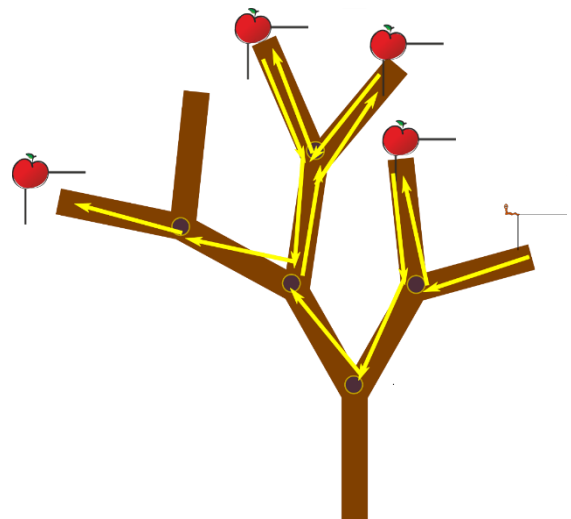
OBJAŠNJENJE

Ako crv želi pojesti sve jabuke, mora prijeći najmanje 13 dijelova grana.

Crv želi pojesti sve jabuke. 4 i 6 nisu točni odgovori jer nije moguće dohvatiti sve jabuke prolazeći samo 4 ili 6 dijelova grana. Putevima duljine 4 i 6 crv će dohvatiti samo jednu jabuku.

Kako bi uspio pojesti sve jabuke, prvo mora dohvatiti najbližu jabuku, a zatim preostale tri jabuke. Primijetimo da nije važno kojim redom će crv dohvatiti preostale jabuke.

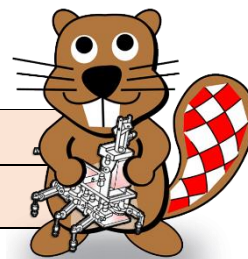
Put s 15 dijelova je predug, stoga put s 13 dijelova omogućuje crvu dohvatiti sve jabuke a da pri tome prijeđe najmanji broj dijelova grana.



RAČUNALNA POVEZANOST

U informatici nije dovoljno planirati naredbe za izvršavanje zadatka, nego je često potrebno pronaći rješenje koje traži najmanji broj koraka. U tom slučaju informatičari kažu da optimiziraju rješenje. Stablo u ovom zadatku predstavlja posebnu vrstu dijagrama u kojem su neke važne točke stabla spojene dijelovima grana. Takve dijagrame zovemo grafovima. Ovaj zadatak je zapravo pronalaženje posebnog puta na grafu, puta koji je najkraći mogući a na kojem će crv iz našeg zadatka pojesti najviše jabuka.

DABROVA BRANA



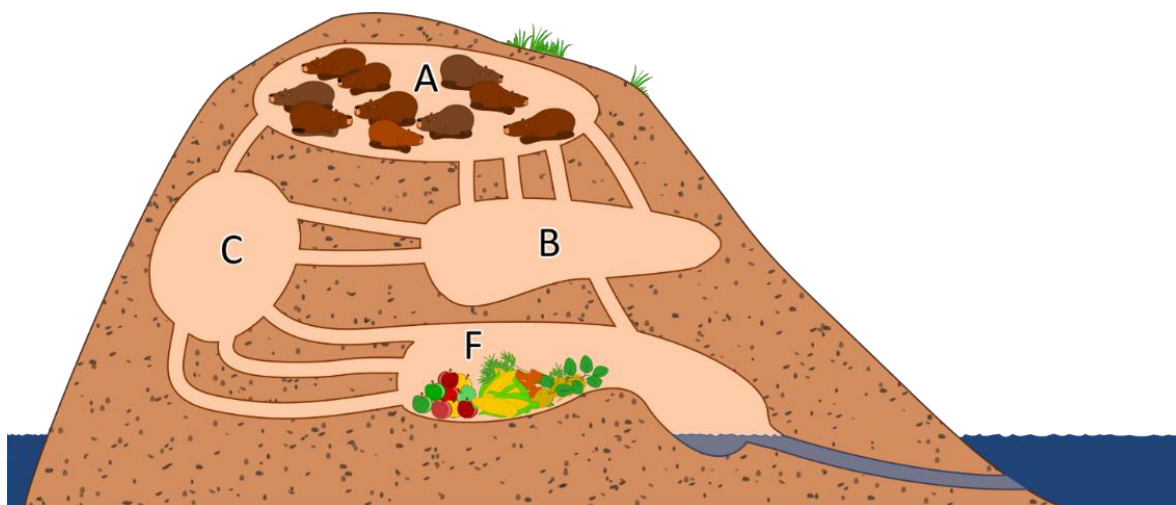
Oznaka zadatka: 2017-CH-07b

Tip pitanja: brojčano

Ključne riječi: graf, planiranje, mreža protoka, teorija grafova, kapacitet

ZADATAK

U dabrovoj brani sobe su spojene tunelima. Prve tri sobe (A, B i C) su dnevne sobe, četvrta soba (F) je spremište hrane (vidi sliku).



U sobi A nalazi se 10 dabrova. Svi su jako gladni i žele čim prije stići do sobe F u kojoj se nalazi spremište hrane. Za prolaz tunelom potrebna je 1 minuta. Samo jedan dabar smije prolaziti tunelom u određenom trenutku. Dva ili više dabrova ne mogu ići tunelom jedan iza drugoga.

Nema ograničeno koliko dabrova može biti u nekoj od soba istovremeno tako da u nekom trenutku u jednoj sobi može biti i svih deset dabrova.

Sobe su povezane određenim brojem tunela:

- između soba A i B: 4 tunela
- između soba A i C: 1 tunel
- između soba B i C: 2 tunela
- između soba B i F: 1 tunel
- između soba C i F: 3 tunela.

U svakoj sobi može biti istovremeno i svih deset dabrova, tj. nije ograničeno koliko dabrova može biti u nekoj od soba istovremeno.

PITANJE/IZAZOV

U najboljem slučaju, nakon koliko minuta će svi dabrovi biti u spremištu hrane.

TOČAN ODGOVOR

Točan odgovor je 4. U najboljem slučaju svi dabrovi će biti u spremištu nakon 4 minute.

OBJAŠNJENJE

U najboljem slučaju svi dabrovi će biti u spremištu nakon 4 minute.

Graf ima 4 najkraća puta, oba puta mogu primiti samo jednog dabra i svakome od njih je potrebno ukupno dvije minute za put:

- od A prema B prema F
- od A prema C prema F

Postoji put kojim mogu proći dva dabra istovremeno, ali je potrebno 3 minute za prolaz:

- od A prema B prema C prema F

Da bi ukupno vrijeme bilo najkraće, put od A prema B mora se koristiti sa smanjenim kapacitetom. Najbolje rješenje je poslati 3 dabra u prvoj minuti i 3 dabra u drugoj minuti.

Budući da samo tri tunela izlaze iz sobe B, ako bismo koristili tunel od A prema B u punom kapacitetu (4 dabra), zbog tunela od B prema C i B prema F, jedan bi dabar ostao čekati u tunelu B.

Sljedeća tablica objašnjava radnje iz minute u minutu sve dok svi dabrovi ne budu u spremištu hrane.

Iako postoji samo jedno najbolje rješenje (4 minute) do njega se može doći na nekoliko načina.

U ovom primjeru dabrovi ne moraju čekati u sobi B.

Putovanje kroz tunele	Broj dabrova u sobi nakon određenog koraka			
	A	B	C	F
Stanje na početku	10	0	0	0
3 dabra idu iz A prema B (smanjeni kapacitet)				
1 dabar ide iz A prema C				
Stanje nakon 1 minute	6	3	1	0
3 dabra idu iz A prema B (smanjeni kapacitet)				
1 dabar ide iz B prema F				
2 dabra idu iz B prema C				
1 dabar ide iz C prema F				
1 dabar ide iz A prema C				
Stanje nakon 2 minute	2	3	3	2
1 dabar ide iz A prema B (izabran najkraći put)				
1 dabar ide iz B prema F				
2 dabra idu iz B prema C				
1 dabar ide iz A prema C (izabran najkraći put)				
3 dabra idu iz C prema F				
Stanje nakon 3 minute	0	1	3	6
1 dabar ide iz B prema F				
3 dabra idu iz C prema F				
Stanje nakon 4 minute	0	0	0	10

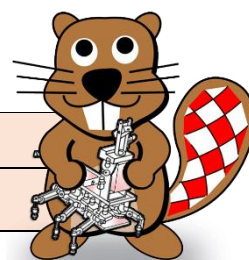
RAČUNALNA POVEZANOST

Mrežu tunela možemo zamisliti kao mrežu protoka u teoriji grafova. To je takozvani usmjereni graf gdje svaki vrh ima kapacitet (maksimalni broj tunela između soba) i svaki vrh prima protok. Iznos protoka na vrhu ne može premašiti kapacitet vrha.

Cilj je optimizirati kretanje dabrova kroz mrežu tunela tako da što veći broj dabrova prođe u što kraćem vremenu. Na primjer, takve se mreže koriste za modeliranje cestovnog prometa. Postoji nekoliko algoritama za ovaj problem, jedan od njih je Ford-Fulkersonov algoritam.

Naš problem je poseban slučaj toga problema jer dozvoljava čekanje u sobama B i C ukoliko nije moguće nastaviti put. U klasičnom mrežnom protoku ništa ne smije stajati na mjestu. Sve što dođe u određeni vrh, mora odmah nastaviti dalje slijedećim bridovima. Nadalje, u klasičnom problemu putevi su unaprijed opisani, dok u našem problemu slobodno biramo kojim putem će dabrovi proći kroz tunele.

ZUBNA PUTOVNICA



Oznaka zadatka: 2017-IE-01

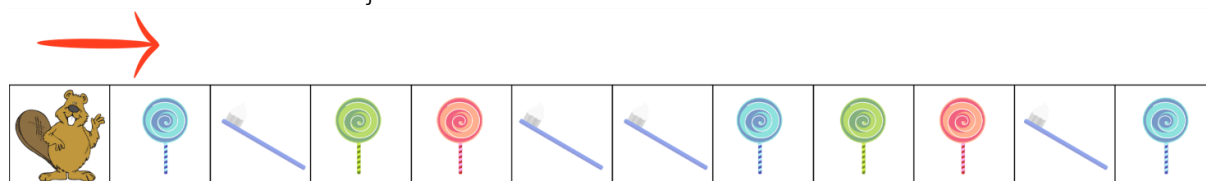
Tip pitanja: brojčani odgovor

Ključne riječi: iteracija, aritmetika

ZADATAK

Dabrić je u hodniku punom lizalica i četkica za zube. Kako bi pojeo lizalice, smije se kretati hodnikom samo pravo. Nije dozvoljeno vraćati se unatrag. Dabrić može prati zube kad god pronade četkicu za zube. Nakon što pojede dvije lizalice, mora oprati zube prije nego pojede sljedeću lizalicu.

U svakom koraku dabrić može pojesti lizalicu **ILI** oprati zube **ILI** nastaviti hodati. Ne može uzeti lizalicu ili četkicu za zube sa sobom na sljedeći korak.



PITANJE/IZAZOV

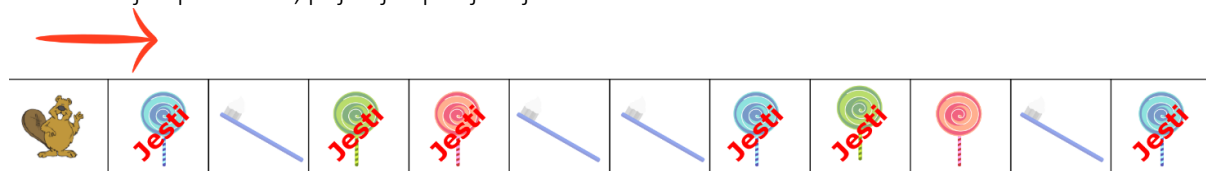
Koliko najviše lizalica dabrić može pojesti, a da mu zubi pri tome budu zdravi?

TOČAN ODGOVOR

Točan odgovor je 6. Dabrić može pojesti 6 lizalica a da mu zubi ostani zdravi.

OBJAŠNJENJE

Dabrić počinje hodati i jesti lizalice. Nakon što je pojeo prvu lizalicu na koju je naišao, oprao je zube. Potom je pojeo obje lizalice te oprao zube. Međutim, kada je naišao na 3 lizalice za redom, jednu je morao preskočiti zbog zadanog uvjeta koji mu nalaže da nakon dvije pojedine lizalice, mora oprati zube. Nakon što je oprao zube, pojeo je i posljednju lizalicu.



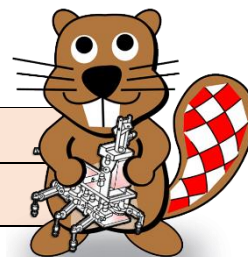
RAČUNALNA POVEZANOST

U ovom zadatku dabar mora pojesti što više lizalica, a to i nije tako jednostavno budući da postoji ograničenje. Ograničenje je pravilo koje nas tjera da nešto napravimo, odnosno, da zadovoljimo određen uvjet.

Na primjer, kada igrate društvene igre, morate poštivati pravila koja za tu igru vrijede. Ponekad su pravila jednostavna, ali mogu biti i složena.

U informatici, ograničenja su važan dio zadatka, ali zbog svoje složenosti, mogu otežati i njegovo rješavanje. Stoga je važno u potpunosti razumjeti pravila kako bismo mogli riješiti zadatak.

PIGPEN ŠIFRA



Oznaka zadatka: 2017-US-01

Tip pitanja: Višestruki odgovor

Ključne riječi: kriptografija, kod, Pigpen šifra

ZADATAK

Dabar Benjamin i njegova susjeda Katarina tajno komuniciraju ostavljajući poruke ispod vrtne ograde. Osmislili su svoje kodove kako bi bili sigurni da nitko drugi neće razumjeti poruke.

Koristili su sljedeći kod:

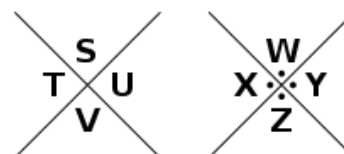
Na primjer, riječ DABAR piše se na sljedeći način:



Benjamin je od Katarine dobio sljedeću poruku:



A	B	C	J	K	L
D	E	F	M	N	O
G	H	I	P	Q	R



PITANJE/IZAZOV

Što je Katarina poručila Benjaminu?

PONUĐENI ODGOVORI

- b) Naći će se kod drva jabuke
- c) Jabuke su bolje od naranče
- d) Za večeru će jesti naranču
- e) Naći će se kod drva naranče

TOČAN ODGOVOR

- a) Za večeru će jesti jabuke

OBJAŠNJENJE

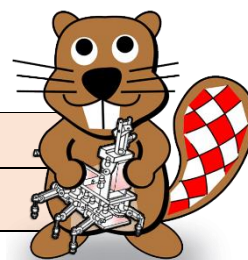
Točan prijevod je «Jabuke za večeru»

RAČUNALNA POVEZANOST

U informatici, enkripcija se koristi za kodiranje poruka tako da je mogu čitati samo ljudi koji znaju na koji način je poruka kodirana.

Postoji mnogo različitih vrsta algoritama za kodiranje. Algoritam za izvršenje kodiranja zove se kod ili šifra: Pigpen šifra, koja se koristi u zadatku, je zamjenska šifra i smatra se prilično slabom jer relativna frekvencija slova ostaje ista u šifriranom i originalnom tekstu. Dakle, tekst se može dešifrirati već samom analizom šifriranog teksta. Kriptografi su osmislili složenije šifre u kojima ne postoji očigledan odnos između originalnog i šifriranog teksta: na primjer, prvi može biti kratak, a drugi dugačak tekst, broj svakog slova u šifri gotovo je slučajan itd.

SKRAĆIVANJE PORUKA



Oznaka zadatka: 2017-CY-05

Tip pitanja: višestruki odabir

Ključne riječi: kodiranje duljine (RLE), komprimiranje podataka

ZADATAK

Dabrica Ana traži skrivene igračke. Svaka igračka označena je velikim slovom abecede (A, B, C, ..., Ž). Ana želi prijatelju poslati kratku poruku o tome koliko je kojih igračaka pronašla. Odlučila je skratiti poruku, i to tako da ispred znaka koji odgovara pronađenoj igrački napiše broj koji označava koliko je tih igračaka pronašla.

Na primjer: umjesto da pošalje poruku AAAAAABBBCCCC Ana će poslati poruku 6A3B5C.

PITANJE/IZAZOV

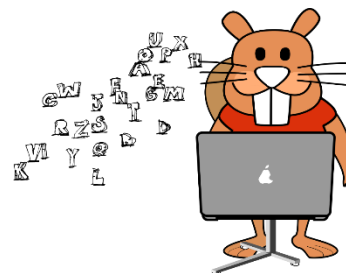
Ako svako slovo i svaka znamenka broja zauzima jedno mjesto, koliko mjesta će zauzeti slijedeća poruka: DDDDDDEEEEEAAAAAABBBBBBBBBBBBBBBBBBCCCCCCCCGGGGGGGGGGHHHH

PONUĐENI ODGOVORI

- a) 15
- b) 16
- c) 17
- d) 18

TOČAN ODGOVOR

- c) 17



OBJAŠNJENJE

Originalna poruka: DDDDDDEEEEEEEEEAAAAAABBBBBBBBBBBBBBBBBBCCCCCCCCGGGGGGGGGGHHHH

Komprimirana poruka: 7D6E6A5B10F8C9G4H

Znakovi: $2+2+2+2+3+2+2+2=17$ (uočiti da se za 10F koriste 3 znaka dok za ostale 2)

RAČUNALNA POVEZANOST

Sažimanje ili komprimiranje podataka je proces smanjenja veličine podatkovne datoteke.

Kod postupka sažimanja "bez gubitaka" sažeta poruka istovjetna je izvornoj bez da je u procesu došlo do gubitaka podataka i kvaliteta informacija. Datoteka sažeta na ovaj način, nakon što se dekomprimira, potpuno odgovara originalnoj datoteci.

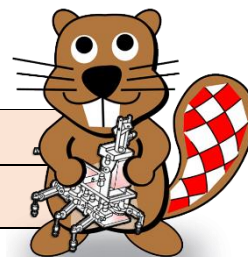
Komprimirane datoteke zauzimaju manje prostora za pohranu i mogu se prenijeti na druga računala brže nego nekomprimirane datoteke.

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

U ovom primjeru upotrebljavamo RLE (Run-Length Encoding) sažimanje podataka bez gubitaka, u kojem se niz jednakih znakova zamjenjuje drugačijim, kraćim zapisom.

Kodiranu poruku pohranjujemo kao vrijednost jednog podataka i broja njegovog pojavljivanja, a ne kao originalni prikaz. RLE je najkorisniji u radu sa podacima koji se ponavljaju, na primjer, u jednostavnim grafičkim prikazima poput ikona, crteža i animacija. S datotekama koje nemaju puno RLE podataka sažimanje nije učinkovito. Umjesto toga može se povećati veličinu datoteke. Dakle, sažimanje je učinkovitije ukoliko se prilagodi vrsti podataka.

NAJSLAĐI ZALOGAJ



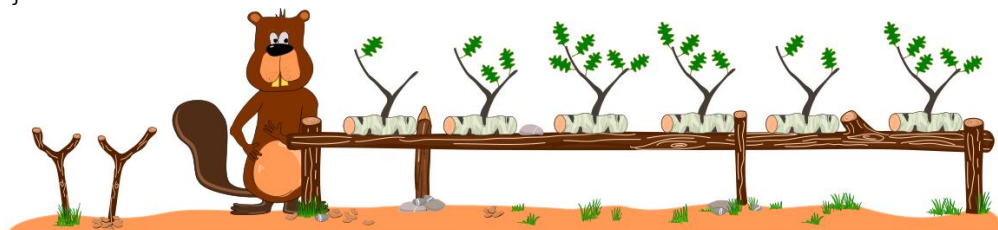
Oznaka zadatka: 2017-CH-08b

Tip pitanja: brojčano

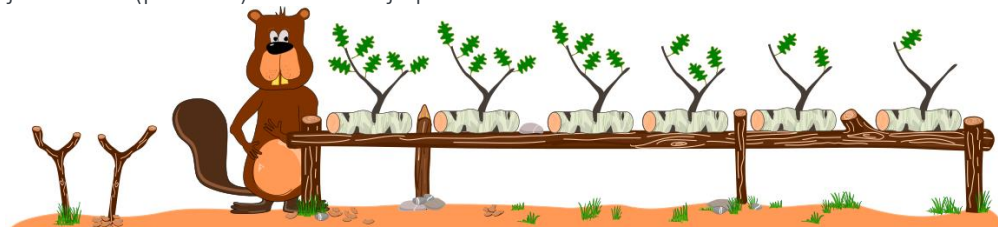
Ključne riječi: sortiranje, zamjena mjesta dvjema varijablama

ZADATAK

Dabar Daniel slaže svoju zimnicu. Njegovo omiljeno deblo je deblo s najviše grančica. To je ujedno i najukusnije deblo.



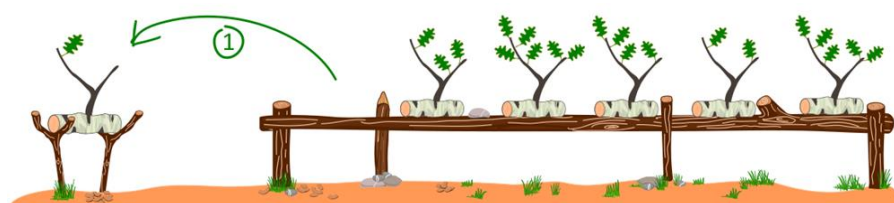
Želja mu je sortirati (posložiti) deblo kako je prikazano na slici.



Za preslagivanje trebat će mu grane pored njega na koje će privremeno odlagati deblo. To je takozvano privremeno spremište. Svako deblo koje odloži u privremeno spremište mora potom vratiti na odgovarajuće mjesto.

PITANJE/IZAZOV

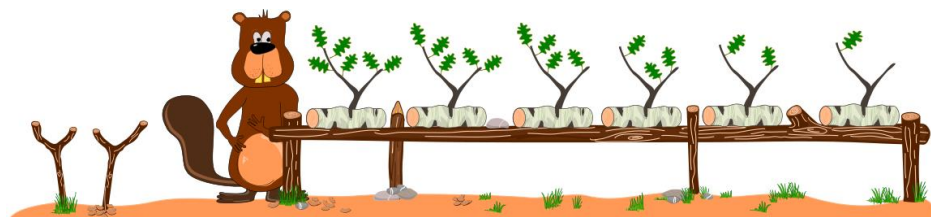
Koliko puta Daniel mora koristiti privremeno spremište ? Pogledaj sliku i razmisli!



TOČAN ODGOVOR

Točan odgovor 4.

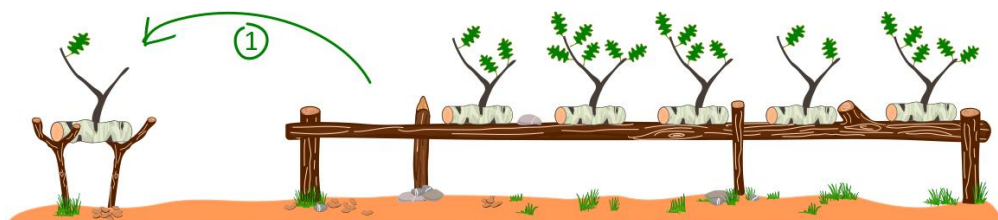
Dabar Daniel mora 4 puta koristiti privremeno spremište kako bi poredao deblo kao što je prikazano na slici.



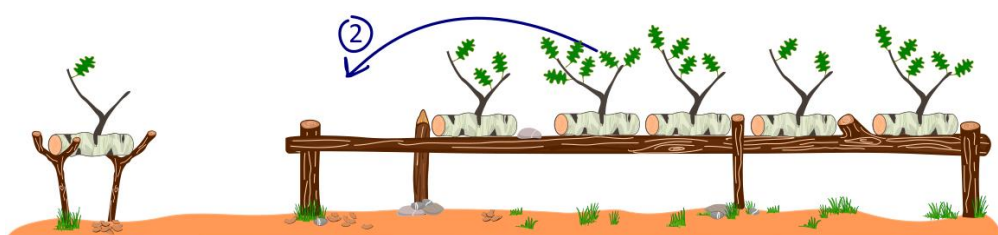
OBJAŠNJENJE

Postoji više načina kako riješiti ovaj problem. Najjednostavniji način je zamijeniti debla pomoću privremenog spremišta:

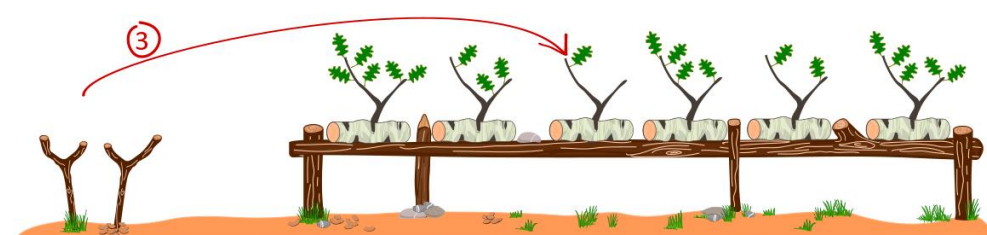
Korak 1: prvo deblo premjestimo u privremeno spremište



Korak 2: pomakni drugo deblo (s najviše grančica) na mjesto prvoga



Korak 3: deblo iz pomoćnog spremnika vrati na slobodno mjesto



Ponavljaj postupak dok ne sortiraš debla.

RAČUNALNA POVEZANOST

Način na koji ćemo zamijeniti dva debla korištenjem privremenog spremišta često se koristi u računalnoj znanosti. Spremnici (spremišta) su obično varijable. Varijable su neizostavan dio svakog programskog jezika i služe kao pomoć u samom procesu programiranja.

Da bismo zamijenili vrijednosti dviju varijabli (nazovimo ih „a“ i „b“) potrebna nam je i privremena varijabla (nazovimo je „T“).

Vrijednost varijable „a“ želimo zamijeniti vrijednošću varijable „b“ pomoću pomoćne varijable „T“ na način da varijabli „T“ dodajemo vrijednost varijable „a“, potom varijabli „a“ dodajemo vrijednost varijable „b“ i na kraju varijabli „b“ dodajemo vrijednost varijable „T“. Privremenu varijablu koristimo od prvog do zadnjeg elementa, kada ju u konačnici zamijenimo sa najmanjim preostalim elementom na popisu.

POPIS AUTORA I SURADNIKA U IZRADI ZADATAKA DABAR 2017

	Andrea Adamoli		Gyöngyi Hidvéginé Kuliga		Chris Roffey
	Nursultan Akhmetov		Sungwoo Huh		Eszter Ruzicska
	Adil Aliyev		Heikki Hyyrö		Morteza Saghafian
	Daumilas Ardickas		Thomas Ioannou		Victor Schmidt
	Haim Averbuch		Svetlana Jakšić		Eljakim Schrijvers
	Wilfried Baumann		Filiz Kaleliolu		Masood Seddighin
	Bartosz Bieganski		Mihaela Kelava		Miko Shimabuku
	Andrej Blaho		Dong Yoon Kim		Taras Shpot
	Daphne Blokhuis		Vaidotas Kincius		Pär Söderhjelm
	Carmen Bruni		Ivana Kosirova		Maciej M. Sysło
	Szu-ching Chen		Lidija Kralj		Hrvoje Šimić
	Sébastien Combéfis		Kryshtopa Marina		Sanja Pavlović Šijanović
	Raluca Constantinescu		Volkan Kukul		Seiichi Tani
	Kris Coolsaet		Regula Lacher		Gergely Tassy
	Valentina Dagiene		En-hsuan Lee		Edit Temesvári
	Darija Dasovic Rakijašić		Violetta Lonati		Narumi Teshima
	Christian Datzko		Milan Lukić		Aliaksei Tolstikau
	Susanne Datzko		Ljubov Jaanuska		Monika Tomcsányiová
	Janez Demšar		Dimitris Mavrovouniotis		Françoise Tort
	Olivier Ens		Karolína Mayerová		Ahto Truu
	Veerle Fack		Hiroki Manabe		Jiří Vaníček
	Lidia Feklistova		Haruki Nakane		Troy Vasiga
	Barnabás Gellér		Henry Ong		Willem van der Vegt
	Ionut Gorgos		Ferhat Kadir Pala		Nicolette Venn
	Arnheiður Guðmundsdóttir		Pedro Paredes		Corina-Elena Vint
	Ahmad Wafiuddin Harun		Wolfgang Pohl		Michael Weigend
	Urs Hauser		Sergej Pozdniakov		Michal Winczer
	Hans-Werner Hein		P. Pretti		Aida Younesi
	Fredrik Heint		Daniel Rakijašić		Khairul Mohamad Zaki
	Juraj Hromkovic		Franc Rosamond		



Vidimo se i sljedeće godine, zar ne? 😊