

Upute natjecateljima

Ovo natjecanje sastoji se od rješavanja **6 problemskih zadataka** u vremenu od **3 sata** (180 minuta). Zadatke je potrebno rješavati u jednom od sljedećih programskih jezika: Pascal, C ili C++. Za natjecanje u Pascalu preporuča se programski alat **Free Pascal**, a za natjecanje u jezicima C i C++ programski alati **DJGPP** i **MinGW (Dev-C++)**. Dozvoljeno je koristiti editor po izboru te pomoću navedenih alata prevoditi izvorni kod u izvršnu datoteku.

Za početak, provjerite da ste dobili od vašeg voditelja **točno 11 stranica** (uključujući ovu koju trenutno čitate). Morate imati dvije stranice s uputama (ovu i još jednu), jednu uvodnu stranicu s tablicom zadataka, i još 8 stranica s tekstovima zadataka.

Tijekom natjecanja **ne smijete komunicirati** ni sa jednom osobom, osim voditelja natjecanja. To znači da morate **raditi samostalno i ne smijete koristiti Internet**. Također, zabranjena je upotreba bilo kakvih ranije napisanih programa ili dijelova programa.

Po isteku vremena predviđenog za natjecanje, morate voditelju natjecanja predati disketu sa snimljenim izvornim kôdovima rješenja. Nakon natjecanja, Sustav za evaluaciju će testirati vaša rješenja na, unaprijed zadanim, test podacima i dodijeliti vam određeni broj bodova. Na kraju svakog zadatka dani su primjeri test podataka. Ti primjeri služe da bi vam tekst zadataka bio što je moguće jasniji te za provjeru formata ulaza i izlaza, a ne služe za provjeru ispravnosti vašeg programa. Ako vaš program radi na tim primjerima, to **nije garancija** da će raditi za ijedan službeni test podatak.

Pripazite na to da ne nosi svaki zadatak jednak broj bodova. Lakše i brže rješivi zadaci nose manje bodova, dok teži nose više bodova. U većini slučajeva svaki test podatak u nekom zadatku nosi jednak broj bodova. Ukupni broj bodova na nekom zadatku je jednak zbroju bodova test podataka koji se poklapaju sa službenim rješenjem. Ukupan broj bodova jednak je zbroju bodova na svim zadacima.

Primijetite da **svi zadaci imaju ulaz sa standardnog ulaza (tipkovnice) i izlaz na standardni izlaz (monitor)**. Sve informacije o zadacima (ime zadatka, ulaz, izlaz, vremensko i memorijsko ograničenje, način bodovanja) možete naći na uvodnoj stranici s naslovom "Zadaci". Ako vam nije jasno nešto u vezi načina provedbe ovog natjecanja, odmah postavite pitanje voditelju, da vam to razjasni.

Tijekom cijelog natjecanja možete postavljati pitanja voditelju natjecanja u vezi zadataka. Dozvoljena su pitanja **koja razjašnjavaju nejasnoće u tekstu zadatka**. Ne smijete postavljati pitanja u vezi rješavanja zadataka. Prije nego postavite pitanje, pročitajte još jednom zadatak, jer je moguće da ste u prethodnom čitanju preskočili dio teksta zadatka.

Upute natjecateljima

VAŽNO za C/C++!

Glavni program (glavna funkcija) **mora** biti deklariran kao: `int main(void) { ... }`.

Program mora završiti svoje izvođenje naredbom `return 0;` unutar funkcije `main` ili naredbom `exit(0);`.

Zabranjeno je koristiti biblioteke `<conio.h>` i `<cconio>`, kao i sve funkcije deklarirane u ovim bibliotekama (npr. `clrscr()`; `getch()`; `getche()`; i sl.). Zabranjeno je koristiti i **sve** sistemske (nestandardne) biblioteke.

Zabranjeno je koristiti funkcije `itoa()` i `ltoa()` jer one ne postoje u standardu jezika C/C++. Umjesto tih funkcija možete koristiti funkciju `sprintf()`, koja ima i veće mogućnosti primjene, deklariranu u `<stdio.h>` i `<cstdio>`.

Zabranjeno je koristiti sve funkcije za rad s datotekama (`fopen()`, `fclose()`, `fprintf()`, `fscanf()`, ...), kao i deklarirati varijable tipa `FILE *`.

Dozvoljeno je koristiti sve ostale standardne biblioteke (koje su dio jezika), uključujući i STL (Standard Template Library) u jeziku C++.

VAŽNO za Pascal!

Program **mora** regularno završiti svoje izvođenje naredbom `end.` unutar glavnog programa ili naredbom `halt;`.

Zabranjeno je koristiti bilo kakve biblioteke, a posebno biblioteku `crt`, tj. zabranjeno je u programu imati direktivu `uses`. To znači da u programu ne smije biti naredbi `clrscr()` i `readkey()`.

Nepoštivanje ovih uputa ili nepridržavanjem formata izlaznih podataka rezultirat će nepovratnim gubitkom bodova. Nemojte ispisivati ništa što se u zadatku ne traži, kao npr. poruke tipa 'Rjesenje je:' ili 'Unesite brojeve' i slično!

Napomena: ista ovakva ograničenja će biti i na ovogodišnjem Državnom natjecanju.

Nakon natjecanja, sva eventualna pitanja, komentare i mišljenja o zadacima uputite na donje e-mail adrese.

Sretno i uspješno natjecanje!

Autori zadataka, test podataka i rješenja
Tomislav Grbin, tomislav.grbin@gmail.com
Luka Kalinovčić, kalinovcic@gmail.com

Zadaci

Zadatak	SKOČIMIŠ	PTICE	MRAVOJED	JEŽ	SKAKAVAC	KRTICA
Izvorni kôd	skocimis.bas skocimis.pas skocimis.c skocimis.cpp	ptice.bas ptice.pas ptice.c ptice.cpp	mravojed.bas mravojed.pas mravojed.c mravojed.cpp	jez.bas jez.pas jez.c jez.cpp	skakavac.bas skakavac.pas skakavac.c skakavac.cpp	krtica.bas krtica.pas krtica.c krtica.cpp
Ulazna datoteka	Standardni ulaz (tipkovnica)					
Izlazna datoteka	Standardni izlaz (zaslon)					
Memorijsko ograničenje	32 MB	32 MB	32 MB	32 MB	35 MB	128 MB
Vremensko ograničenje (po test podatku)	1 sekunda	1 sekunda	1 sekunda	1 sekunda	4 sekunde	3 sekunde
Broj test podataka	10	10	10	10	10	10
Broj bodova (po test podatku)	3	4	7	10	12	14
Ukupno bodova	30	40	70	100	120	140
	500					

Napomena: Vremensko ograničenje je mjereno na računalu baziranom na dva procesora Intel(R) Pentium(R) 4 CPU 3.00GHz i operativnim sustavom Linux. Programe u Pascalu treba kompajlirati s opcijama: `-So -O1 -XS`, a programe u C-u i C++-u s opcijama: `-O2 -lm -static`. U programskom jeziku QBasic dozvoljeno vrijeme izvršavanja programa je dvostruko veće.

1. SKOČIMIŠ

Tri skočimiša igraju se u pustinji. Oni stoje na pravcu i međusobno se preskaču, i to tako da jedan od dva vanjska skočimiša skoči na cjelobrojnu poziciju između preostala dva. Ni u jednom trenutku se dva skočimiša ne nalaze na istoj poziciji.

Vaš zadatak je pomoći im da se igraju što dulje.

Ulazni podaci

U prvom retku nalaze se prirodni brojevi A , B i C ($0 < A < B < C < 100$), početne pozicije skočimiševa na pravcu.

Izlazni podaci

Ispisati koliko najviše skokova mogu napraviti skočimiševi prije kraja igre.

Primjeri test podataka

ulaz

2 3 5

izlaz

1

ulaz

3 5 9

izlaz

3

2. PTICE

Adrian, Bruno i Goran pošli su se učlaniti u klub ljubitelja ptica. Međutim nisu znali da svi članovi moraju proći prijemni ispit. Prijemni ispit sastoji se od N pitanja, a za svako pitanje ponuđena su tri odgovora označena slovima A, B i C.

Kako naši junaci ne znaju ništa o pticama odlučili su pogađati odgovore. Svaki od njih ima svoju teoriju koji niz odgovora je najbolje ponuditi.

Adrian tvrdi da je najbolje pogađati redom: A, B, C, A, B, C, A, B, C, A, B, C...

Bruno tvrdi da je bolje odgovarati redom: B, A, B, C, B, A, B, C, B, A, B, C...

Goran je pak uvjeren da je najbolji niz: C, C, A, A, B, B, C, C, A, A, B, B...

Napiši program koji će na temelju točnih odgovora na zadana pitanja odrediti koji je od njih trojice bio u pravu – **čiji niz odgovora sadrži najviše točnih**.

Ulazni podaci

U prvom redu nalazi se prirodni broj N ($1 \leq N \leq 100$), broj pitanja u ispitu.

U drugom redu nalazi se niz od N znakova 'A', 'B' ili 'C'. Niz predstavlja redom točne odgovore na pitanja postavljena u ispitu.

Izlazni podaci

U prvi red izlaza ispišite cijeli broj M , najveći postignuti broj točnih odgovora.

U sljedeće redove ispišite imena osoba (abecednim redom ako ih je više) čiji nizovi sadrže M točnih odgovora, svako u zaseban red.

Primjeri test podataka

ulaz

5
BAACC

izlaz

3
Bruno

ulaz

9
AAAABBBBB

izlaz

4
Adrian
Bruno
Goran

3. MRAVOJED

Arheolozi su nedavno na lokaciji iza brda pronašli ostatke grčko-rimskih zgrada. Lokaciju možemo podijeliti na $R \cdot S$ kvadratnih polja. Za svako od tih polja arheolozi znaju je li se na njemu nalazila zgrada.

Detaljnim proučavanjem pronađenih predmeta, zaključili su da se radi o **dvije zgrade** iz različitih razdoblja, te da su **tlocrti obaju zgrada kvadrati**.

Kako su zgrade iz različitih razdoblja, moguće je da im se tlocrti preklapaju. Odredite mogući položaj i veličinu (duljinu stranice kvadrata koji predstavlja tlocrt) za obje zgrade.

Ulazni podaci

U prvom redu nalaze se dva prirodna broja R ($1 \leq R \leq 100$) i S ($1 \leq S \leq 100$), dimenzije nalazišta.

U sljedećih R redova nalazi se po S znakova '.' (točka) ili 'x' (malo slovo x). Znak '.' označava polje na kojem nije pronađeno ništa, a znak 'x' da je na tom mjestu bila bar jedna zgrada.

Izlazni podaci

Za svaku od dvije zgrade ispiši red i stupac u kojem se nalazi njen gornji lijevi kut te duljinu stranice zgrade.

Napomena: Rješenje će uvijek postojati i ne mora biti jedinstveno.

Primjeri test podataka

ulaz

3 3

xx.

xxx

...

izlaz

1 1 2

2 3 1

ulaz

4 6

xx....

xx.xxx

...xxx

...xxx

izlaz

1 1 2

2 4 3

ulaz

5 5

.....

xxx..

xxxx.

xxxx.

.xxx.

izlaz

2 1 3

3 2 3

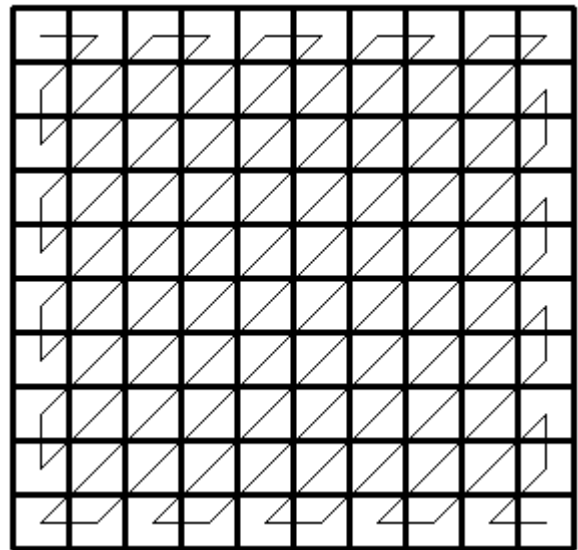
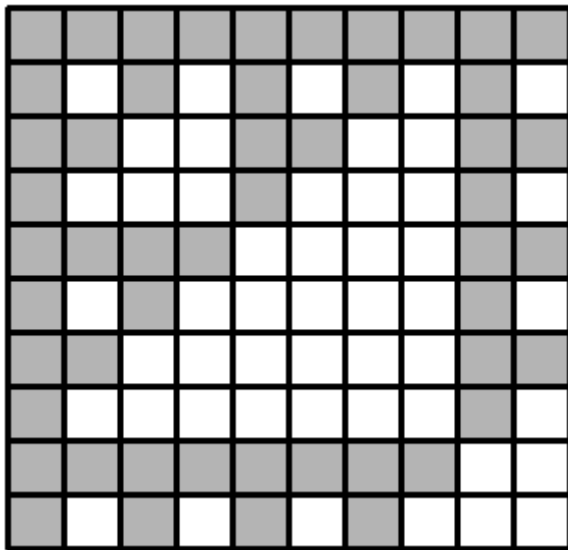
4. JEŽ

Luka je u svom podrumu pronašao sasvim neobičnu ploču. Ona se sastoji od $R \cdot S$ kvadrata poredanih u R redova i S stupaca. Redovi su označeni brojevima od 0 do $R-1$ odozgo prema dolje, a stupci brojevima od 0 do $S-1$ s lijeva na desno. Ono što ploču čini neobičnom je način na koji su kvadrati obojani.

Svaki kvadrat je obojan jednom od dvije boje, bijelom ili sivom, i to tako da je kvadrat:

- bijele boje – ako oznake reda i stupca u binarnom zapisu imaju barem jednu jedinicu na istoj poziciji,
- sive boje – inače.

Lijeva slika prikazuje ploču veličine $10 \cdot 10$.



Lukin jež Damir voli se šetati po toj neobičnoj ploči i on to čini na neobičan način. Jež svoju šetnju započinje na kvadratu s oznakom $(0,0)$, a nastavlja na “cik-cak” način prikazan na desnoj slici. Dok se Damir šeće, Luka **broji koliko je sivih kvadrata** Damir prošetao.

Nakon što prijede K kvadrata, Damir se umori i zaspe. Tada i Luka odlazi u krevet, sretan što je prebrojao sve sive kvadrate kojima se Damir prošetao.

Napiši program koji će na temelju brojeva R , S i K prebrojati sive kvadrate kojima se Damir prošetao.

Ulazni podaci

U prvom redu nalaze se dva cijela broja R ($1 \leq R \leq 1\,000\,000$) i S ($1 \leq S \leq 1\,000\,000$), veličina ploče.

U drugom redu nalazi se prirodni broj K ($1 \leq K \leq R \cdot S$), ukupni broj kvadrata kojima se jež Damir prošetao. Primijetite da broj K ne mora stati u 32-bitni cjelobrojni tip podataka.

Izlazni podaci

U prvi i jedini red ispiši broj sivih kvadratića kojima se jež Damir prošetao.

4. JEŽ

Bodovanje

U test podacima vrijednim 50% bodova, broj K bit će manji od 1 000 000.

Primjeri test podataka

ulaz

10 10
6

izlaz

5

ulaz

3 5
11

izlaz

8

ulaz

10 10
100

izlaz

51

5. SKAKAVAC

Skakavac Zdravko nalazi se u polju cvijeća. Polje sadrži $N \cdot N$ cvjetova poredanih u N redova i N stupaca označenih brojevima od 1 do N . Za svaki cvijet u polju poznato je koliko ima latica.

Zdravko se trenutno nalazi na cvijetu u R -tom redu i S -tom stupcu, a cilj mu je posjetiti što je moguće više cvjetova pritom poštujući sljedeća pravila:

1. Zdravko može skočiti samo u susjedni red ili stupac. Ako skače u susjedni red tada mora preskočiti barem dva stupca, a ako skače u susjedni stupac tada mora preskočiti barem dva reda.

Drugim riječima, Zdravko može skočiti s cvijeta (r_1, s_1) na cvijet (r_2, s_2) ako vrijedi:

- $|r_1 - r_2| = 1$ i $|s_1 - s_2| > 1$ ili
- $|s_1 - s_2| = 1$ i $|r_1 - r_2| > 1$.

2. Broj latica na cvijetu na koji Zdravko skače mora biti strogo veći od broja latica na cvijetu s kojeg dolazi.

Napiši program koji će izračunati najveći mogući broj cvjetova koje Zdravko može posjetiti.

Ulazni podaci

U prvom redu ulaza nalazi se prirodni broj N ($1 \leq N \leq 1500$), veličina polja.

U drugom redu ulaza nalaze se prirodni brojevi R ($1 \leq R \leq N$) i S ($1 \leq S \leq N$), koordinate cvijeta na kojem se nalazi Zdravko.

U sljedećih N redova nalazi se po N cijelih brojeva manjih od 1 000 000 odvojenih razmacima, brojevi latica na cvijeću.

Izlazni podaci

U jedini red izlaza ispiši jedan cijeli broj – najveći mogući broj cvjetova koje Zdravko može posjetiti.

Bodovanje

U test podacima vrijednim 50% bodova, broj N bit će manji od 100.

U test podacima vrijednim 80% bodova, broj N bit će manji od 1000.

Primjeri test podataka

ulaz

```
4
1 1
1 2 3 4
2 3 4 5
3 4 5 6
4 5 6 7
```

izlaz

```
4
```

ulaz

```
5
3 3
20 16 25 17 12
11 13 13 30 17
15 29 10 26 11
27 19 14 24 22
23 21 28 18 13
```

izlaz

```
21
```

6. KRTICA

Krtica Štefica marljiva je i uredna. Ona svoju podzemnu rezidenciju voli držati u najboljem mogućem redu, tako da svaki njen stanar zna gdje mu je što i kako doći gdje želi.

U tu svrhu, sobe je spojila hodnicima tako da se iz svake sobe može doći u svaku drugu na jedinstven način. Udaljenost dvoje sobe računamo kao broj hodnika kojima je potrebno proći na putu iz jedne sobe u drugu.

Unatoč Štefičinom gostoprimstvu, neki njeni nezahvalni gosti se žale da im je predaleko putovati njenim načinom od nekih soba do nekih drugih.

Zato je Štefica odlučila zatvoriti jedan hodnik i izgraditi neki novi i to tako da **udaljenost dvije najudaljenije sobe** budu što je **manja** moguća, a da se još uvijek može doći iz svake sobe u svaku drugu.

Napiši program koji će pronaći koji hodnik treba zatvoriti, a koji izgraditi kako bi nakon toga udaljenost dvije najudaljenije sobe bila što je manja moguća.

Ulazni podaci

U prvom redu nalazi se prirodni broj N ($1 \leq N \leq 300\,000$), broj soba. Sobe su označene brojevima od 1 do N .

U sljedećih $N-1$ redova nalaze se po dva prirodna broja odvojena razmakom, oznake soba između kojih postoji hodnik.

Izlazni podaci

U prvi red izlaza ispišite udaljenost dvije najudaljenije sobe nakon preuređenja rezidencije.

U drugi red izlaza ispišite oznake soba između kojih treba zatvoriti hodnik, odvojene razmakom.

U treći red izlaza ispišite oznake soba između kojih treba izgraditi novi hodnik, odvojene razmakom.

Napomena: Rješenje ne mora biti jedinstveno.

Bodovanje

U test podacima vrijednim 40% bodova, broj N bit će manji od 30.

U test podacima vrijednim 70% bodova, broj N bit će manji od 3000.

Ukoliko je prvi red izlaza vašeg programa točan, a preostala dva nisu, dobit ćete 70% vrijednosti test podatka.

6. KRTICA

Primjeri test podataka

ulaz

4
1 2
2 3
3 4

izlaz

2
3 4
4 2

ulaz

7
1 3
2 3
2 7
4 3
7 5
3 6

izlaz

3
2 3
7 3