

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++. Natjecatelji u Pascalu moraju programirati u programskom alatu **Free Pascal**, a natjecatelji u C-u i C++-u u programskom alatu **DJGPP**. Dozvoljeno je koristiti editor po izboru te pomoću navedenih alata prevoditi izvorni kôd u izvršnu datoteku.

Za početak, provjerite da ste od vašeg voditelja dobili **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 6 stranica s tekstovima zadataka.

Tijekom natjecanja **ne smijete komunicirati** ni sa jednom osobom, osim voditelja natjecanja. To znači da morate **raditi samostalno** i da **ne smijete koristiti mrežu odnosno 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 potpune 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. Svaki test podatak u nekom zadatku nosi jednak broj bodova. Ukupni broj bodova na nekom zadatku jednak je zbroju bodova test podataka koji se poklapaju sa službenim rješenjem. Ukupan broj bodova jednak je zbroju bodova na svim zadacima.

Primjetite da **svi zadaci imaju ulaz sa standardnog ulaza (tipkovnice) i izlaz na standardni izlaz (monitor)**. Sve informacije o zadacima (ime zadatka, ulaz, izlaz, memorijsko i vremensko 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 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 npr. ne smije biti naredbi `clrscr()` i `readkey()`.

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++.

Nepoštivanje ovih uputa ili nepridržavanje 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!

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

Sretno i uspješno natjecanje!

Luka Kalinović i Lovro Pužar
kalinovic@gmail.com, lovro@hsin.hr

Zadaci

Zadatak	PRASE	MAGIJA	MARATON	KAMEN	V	PROSTOR
Izvorni kôd	prase.bas prase.pas prase.c prase.cpp	magija.bas magija.pas magija.c magija.cpp	maraton.bas maraton.pas maraton.c maraton.cpp	kamen.bas kamen.pas kamen.c kamen.cpp	v.bas v.pas v.c v.cpp	prostor.bas prostor.pas prostor.c prostor.cpp
Ulaz	standardni ulaz (tipkovnica)					
Izlaz	standardni izlaz (zaslon)					
Memorija (heap)	32 MB	32 MB	32 MB	32 MB	32 MB	32 MB
Memorija (stack)	8 MB	8 MB	8 MB	8 MB	8 MB	8 MB
Vrijeme (po test podatku)	1 sekunda	1 sekunda	1 sekunda	1 sekunda	1 sekunda	2 sekunde
Broj test podataka	5	10	10	10	14	10
Broj bodova (po test podatku)	4	3	3	5	5	10
Ukupno bodova	20	30	30	50	70	100
	300					

Napomena: Vremensko ograničenje je mjereno na računalu baziranom na dva procesora Intel Pentium 4 3.0 GHz i na operativnom sustavu Linux. Za kompajliranje programa na sustavu se koriste sljedeće opcije:

- C: `-O2 -s -static -std=c99 -lm`
- C++: `-O2 -s -static -lm`
- Pascal: `-O1 -XS`

1. PRASE

N djece sjedi za stolom s majkom i jede ručak. Djeca redom uzimaju komade hrane s prepunog stola.

Neka djeca još nisu naučila osnove pristojnog ponašanja za stolom, pa besramno grabe hranu umjesto da daju priliku i drugima da uzmu. Ako u nekom trenutku dijete uzme komad hrane, a da je **dotad** to dijete već **bilo uzelo više hrane nego sva ostala djeca zajedno**, tad ga majka opomene i kaže mu da je prase.

Zadan je redoslijed kojim djeca uzimaju hranu sa stola. Odredite koliko puta majka kaže nekom djetetu da je prase.

Ulazni podaci

U prvom redu nalazi se prirodni broj N ($1 \leq N \leq 100$), broj komada hrane koji su uzela sva djeca.

Svaki od sljedećih N redova sadrži ime djeteta koje je uzelo jedan komad hrane. Imena će biti nizovi od po najviše 20 malih slova engleske abecede.

Izlazni podaci

U jedan red ispišite koliko puta majka kaže nekom djetetu da je prase.

Primjeri test podataka

ulaz

4
mirko
stanko
stanko
stanko

izlaz

1

ulaz

17
a
b
b
a
a
a
c
a
b
b
c
b
b
b
b
b
b
b

izlaz

4

2. MAGIJA

Dobro poznati mađioničar Al'Dimi Kartimi treba program koji će mu pomoći u dizajniranju stražnje strane njegovih karata.

Stražnja strana karte nastaje tako da Al'Dimi nacrtá gornju lijevu četvrtinu karte, koju prvo **preslika na gornju desnu četvrtinu karte** osnosimetrično s obzirom na sredinu karte, a zatim **gornju polovicu karte preslika na donju polovicu**.

Nakon preslikavanja, Al'Dimi također na svaku kartu dodaje malu nepravilnost (promijeni jedno polje) koja će mu pomoći da otkrije o kojoj je karti riječ (to jest, da vara).

Pomozite Al'Dimiju i napravite program koji za **zadanu gornju lijevu četvrtinu karte** te poziciju nepravilnosti iscrtava cijelu kartu.

Tri primjera Al'Dimijevih karata (nepravilnost je osjenčana):

###.##.###	#.#..#.#	■#.##.##
#####	#.####.#	#.#..#.#
.#####.	#.####.#
..#####.#..#..
#####■#####	.#.#.#.#	..#..#..
#####	.#.#.#.#
..#####.	#.#..#.#
.#####.	#.####.#	##.##.##
#####	#.####.#	
###.##.###	#.#.■#.#	

Ulazni podaci

U prvom redu nalaze se dva prirodna broja, R i S ($1 \leq R, S \leq 50$), broj redaka i stupaca gornje lijeve četvrtine karte.

U svakom od sljedećih R redaka nalazi se po S znakova '.' ili '#', dizajn gornje lijeve četvrtine karte.

U sljedećem retku nalaze se dva prirodna broja, A i B ($1 \leq A \leq 2R, 1 \leq B \leq 2S$), redak i stupac nepravilnosti.

Izlazni podaci

Ispišite stražnju stranu karte koja se dobiva od zadane gornje lijeve četvrtine i za zadanu poziciju nepravilnosti.

2. MAGIJA

Primjeri test podataka

ulaz

2 2
#.
.#
3 3

izlaz

#..#
.##.
.#..
#..#

ulaz

3 3

1 4

izlaz

###.##

#####

ulaz

5 4
#.#.
#.#
#.#
#.#
....
.#.#
10 5

izlaz

#.#.#.#
#.#####
#.#####
#.#.#.#
.....
.#.##.#.
.#.##.#.
.....
#.#####
#.#####
#.#.#.#

3. MARATON

Albert, Barbara, Casper, Dinko, Eustahije počinju maratonsku igru križić-kružić. Igra se na ploči koja se sastoji od $N \times N$ polja.

Na početku igre je svako polje prazno te igrači redom upisuju prvo slovo svojeg imena u neko od praznih polja (budući da je društvo vrlo elitno, nikoja dva igrača nemaju isto prvo slovo imena).

Igra završava kada jedan igrač **smjesti 3 svoja slova uzastopno** u nekom retku, stupcu ili na nekoj dijagonali. Taj igrač zatim uz mnogo buke biva proglašen pobjednikom.

Napišite program koji za zadano stanje ploče određuje je li igra gotova i, ako je, tko je pobijedio.

Ulazni podaci

U prvom redu nalazi se broj N ($1 \leq N \leq 30$), veličina ploče.

U svakom od sljedećih N redaka nalazi se po N znakova. Znakovi mogu biti velika slova engleske abecede (koja predstavljaju slovo upisano u jedno polje) ili '.' (ukoliko je polje prazno).

Ulazni podaci će biti takvi da će postojati najviše jedan pobjednik.

Izlazni podaci

Ako je igra završena, ispišite prvo slovo imena pobjednika. Ako nije, ispišite poruku "jos traje".

Primjeri test podataka

ulaz

3
XOC
XOC
X..

izlaz

X

ulaz

4
....
..A.
AAB.
.B.B

izlaz

jos traje

ulaz

3
ABB
AAA
BBA

izlaz

A

4. KAMEN

Domeniko već skoro dva tjedna leži u svojem vrlo udobnom krevetu jer mu je prijatelj Nedjeljko slučajno bacio golemi kamen na lijevu nogu. Kako je Domeniko već odavno porješavao sve zadatke sa svih DMIH-ova od 1998. godine nadalje, primoran je smisliti novu zanimaciju za ubijanje dosade.

Igra koju je Domeniko smislio igra se na ploči veličine $R \times S$. Na početku igre svako polje na ploči može biti prazno ili blokirano zidom. Domeniko ubacuje kamen u ploču tako da ga stavi u najviši redak nekog stupca, te prepusti gravitaciji da obavi ostatak posla.

Gravitacija djeluje na kamen na sljedeći način:

- Ako se na polju ispod kamena nalazi zid ili ako se kamen nalazi u najnižem retku onda kamen ostaje na mjestu i više se ne pomiče.
- Ako je polje ispod kamena prazno onda se kamen pomiče za jedno polje prema dolje.
- Ako se na polju ispod kamena nalazi drugi kamen onda kamen može skliznuti u stranu i to:
 - Ako su polja lijevo i dolje-lijevo od kamena prazna onda kamen sklizne ulijevo, tj. pomakne se za jedno polje prema lijevo.
 - Ako kamen ne sklizne ulijevo, a ako su polja desno i dolje-desno od kamena prazna onda kamen sklizne udesno, tj. pomakne se za jedno polje prema desno.
 - Ako kamen ne sklizne niti ulijevo niti udesno onda ostaje na mjestu i više se ne pomiče.

Domeniko neće ubaciti sljedeći kamen dok se prethodno ubačeni kamen ne smiri na ploči.

Napišite program koji će nacrtati konfiguraciju ploče nakon što Domeniko ubaci svo kamenje ako znamo redom u koje je stupce Domeniko ubacivao kamenje.

Napomena: Domeniko nikad neće probati ubaciti kamen u do kraja popunjen stupac.

Ulazni podaci

U prvom redu nalaze se prirodni brojevi R i S ($1 \leq R \leq 30000$, $1 \leq S \leq 30$), dimenzije ploče.

U svakom od sljedećih R redova nalazi se po S znakova koji predstavljaju početnu konfiguraciju ploče. Točka predstavlja prazno polje, dok veliko slovo X predstavlja polje blokirano zidom.

U sljedećem redu nalazi se prirodni broj N ($1 \leq N \leq 100000$), broj ubačenog kamenja.

U svakom od sljedećih N redova nalazi se po jedan prirodni broj manji ili jednak od S , redni broj stupca u koji je kamen ubačen.

Napomena: U 60% test podataka će R biti najviše 30.

Izlazni podaci

U R redaka ispišite po S znakova koji predstavljaju konačnu konfiguraciju ploče. Točka predstavlja prazno polje, veliko slovo X predstavlja polje blokirano zidom, dok veliko slovo O predstavlja polje na kojem se nalazi kamen.

4. KAMEN

Primjeri test podataka

ulaz

5 4

.....

.....

X...

.....

.....

4

1

1

1

1

izlaz

.....

O...

X...

.....

OOO.

ulaz

7 6

.....

.....

...XX.

.....

.....

.XX...

.....

6

1

4

4

6

4

4

izlaz

.....

...O...

...XX.

.....

.OO...

.XX...

O..O.O

Pojašnjenje prvog primjera: prvi ubačeni kamen stane na zidu u prvom stupcu. Drugi kamen padne na prvi, sklizne desno te završi na dnu drugog stupca. Treći kamen padne na prvi pa na drugi kamen, sklizne lijevo, te završi na dnu prvog stupca. Četvrti kamen padne na prvi pa na drugi, a zatim sklizne desno.

5. V

Zvonko se opet igra sa znamenkama, iako ga je majka upozorila da se previše bavi matematikom i da bi trebao ići van igrati se s prijateljima.

U svojoj najnovijoj igri, Zvonko traži **višekratnike** nekog prirodnog broja X koji se **sastoje samo od zadanih znamenaka**.

Kako bi Zvonku upropastila zabavu (odgojna mjera), majka je odlučila nabaviti program koji rješava zadani problem. Napišite program koji određuje koliko višekratnika broja X se nalazi između brojeva A i B (uključivo), a da u dekadskom zapisu sadrže samo zadane znamenke.

Ulazni podaci

U prvom redu ulaza nalaze se tri prirodna broja X , A , i B ($1 \leq X < 10^{11}$, $1 \leq A \leq B < 10^{11}$).

U drugom redu nalazi se niz dekadskih znamenaka koje Zvonko smije koristiti pri slaganju višekratnika. Znamenke će biti zadane bez razmaka, sortirane uzlazno i bez ponavljanja.

Izlazni podaci

U jedini red ispišite broj višekratnika koje Zvonko može napraviti.

Primjeri test podataka

ulaz

2 1 20
0123456789

izlaz

10

ulaz

6 100 9294
23689

izlaz

111

ulaz

5 4395 9999999999
12346789

izlaz

0

6. PROSTOR

U nekom dalekom prostoru, u nekom dalekom vremenu, bezbrižno je živjelo pleme pravokutnika. Pravokutnici su živjeli duhovnim životom, paralelni s nekom od koordinatnih ravnina.

Jednoga dana, u njihov mali svijet ušetao se kvadar, mirno jašući na ikozaedru, šepureći se svojim oštrim bridovima i pozitivnim volumenom. Svi pravokutnici su sa divljenjem gledali u njega i maštali. Od toga dana više ništa nije bilo kao prije. Pravokutnici su se počeli međusobno uspoređivati po površini, opsegu, pa čak i po omjeru duljina stranica.

I tako je jednoga dana došlo do prvog sukoba oko vlasništva zajedničkih točaka. S vremenom, **svaka dva pravokutnika koja dijele bar jednu zajedničku točku** (ovo uključuje i pravokutnike koji se samo dodiruju) su se sukobila i postala smrtni neprijatelji.

Na vama je da vratite mir u ovu malu zajednicu, te da pomirite svaki par sukobljenih pravokutnika. Izračunajte **koliko parova** pravokutnika treba pomiriti.

Ulazni podaci

U prvom redu ulaza nalazi se prirodni broj N ($1 \leq N \leq 100\,000$), broj pravokutnika.

U svakom od sljedećih N redova ulaza nalazi se po 6 prirodnih brojeva odvojenih po jednim razmakom. Prva tri broja predstavljaju koordinate jedne točke pravokutnika, a druga tri broja koordinate njoj nasuprotne točke pravokutnika.

Koordinate su prirodni brojevi između 1 i 999 (uključivo).

Svaki pravokutnik je paralelan s jednom od koordinatnih ravnina, tj. u točno jednoj od tri dimenzije dvije odgovarajuće koordinate će biti jednake.

Izlazni podaci

U jedini red ispišite ukupan broj sukobljenih pravokutnika.

Primjeri test podataka

ulaz

```
3
1 1 1 1 3 3
1 3 3 1 6 6
1 4 4 1 5 5
```

izlaz

```
2
```

ulaz

```
3
15 10 10 15 20 20
10 15 10 20 15 20
10 10 15 20 20 15
```

izlaz

```
3
```

ulaz

```
5
4 4 5 4 3 2
5 3 2 4 3 1
5 4 3 1 1 3
1 4 3 1 5 4
5 5 4 5 4 2
```

izlaz

```
4
```