



Infokup

Županijsko natjecanje / Algoritmi /
Srednja škola (II. Podskupina)



Agenција za odgoj i obrazovanje
Education and Teacher Training Agency



MINISTARSTVO ZNANOSTI, OBRAZOVANJA
I ŠPORTA REPUBLIKE HRVATSKE

udruga mladih programera
dump

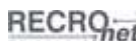


Sponzori Infokupa

Glavni sponzor



Mali sponzori



Medijski pokrovitelji



Microsoft

Microsoft Innovation Center
Ljubljana

Microsoft Innovation Center
Vrhnika



Sadržaj

Upute za natjecatelje	2
Primjer pravilno napisanog programa	3
Zadaci	4
Zadatak: Kvadrat	5
Zadatak: Palindrom	6
Zadatak: Stabla	7
Zadatak: DNA	9



Upute za natjecatelje

Kod svakog pojedinog zadatka obratite pozornost na poglavlja **ulazni podaci** i **izlazni podaci**. Tu su definirana pravila vezana za format ulaznih i izlaznih podataka koji mora biti strogo poštovan kako bi vaša rješenja bila točno evaluirana. Za ilustraciju i bolje razumijevanje pogledajte poglavlje *primjer pravilno napisanog programa*.

Ulaz i izlaz treba se odvijati preko standardnog ulaza i standardnog izlaza (to znači **cin**, **cout**, **printf** i/ili **scanf**). Vaš program sa standardnog ulaza (**cin** ili **scanf**) mora očekivati samo ulazne podatke, a na standardni izlaz (**cout** ili **printf**) mora ispisivati samo izlazne podatke. Ako vaš program bude čekao na unos nečeg drugog osim ulaznih podataka ili ispisivao nešto drugo osim izlaznih podataka (npr. "Unesite brojeve...", "Rješenje je..." i slično), nećete dobiti bodove za taj zadatak, jer evaluator to ne očekuje. U pisanju programa dozvoljeno je koristiti samo standardne biblioteke, pa je tako primjerice zabranjeno koristiti naredbe **clrscr()**; ili **getch()**; jer su one dio **conio.h** biblioteke koja nije standardna u jeziku **C/C++**.

Važno je napomenuti da ne trebate kreirati izvršnu datoteku (.exe), jer će ju sustav sam kreirati iz izvornog koda na sljedeći način:

- **pascal**: `fpc -O1 -XS -oPRG.exe PRG.pas`
- **C**: `gcc -o PRG.exe PRG.c -std=c99 -O2 -s -static -lm`
- **C++**: `g++ -o PRG.exe PRG.cpp -O2 -s -static -lm`

(gdje je PRG ime programa)

Računalo na kojem se izvode programi i mjerena su vremenska ograničenja je Linux računalo s 2GHz radnog takta procesora.

Vaš program treba biti pisan u programskom jeziku C, C++ ili Pascal i mora regularno završiti svoje izvođenje. Program se treba izvršiti do kraja tj. do **return 0;** na kraju funkcije 'main' koja treba biti deklarirana kao **int main()**, ili naredbom **exit(0);**. Pogledajte priložene primjere. U programskom jeziku pascal program se treba izvršiti do kraja tj. Do **'end.'** ili naredbom **halt(0);**. Vaši programi **ne smiju pristupati** nikakvim datotekama **niti ih kreirati**, kršenje ovog pravila rezultirati će gubitkom bodova za taj zadatak. Bilo kakav pokušaj **pristupanja sistemskim datotekama računala na kojem se nalazi evaluator ili pokušaj upravljanja tim računalom** rezultirat će **diskvalifikacijom** tog natjecatelja.

Za dodjelu bodova važan je samo točan ispis rezultata. Prilikom evaluacije nitko neće gledati vaš izvorni kôd već će on samo biti korišten za izradu izvršne datoteke, a bodove za pojedini test podatak će dobiti samo oni programi koji budu generirali **točan rezultat unutar navedenog vremena i memorijskog ograničenja**.

Obratite pažnju da svi zadaci ne nose jednak broj bodova. Lakši i brže rješivi zadaci nose manje bodova, a teži zadaci za čije je rješavanje potrebno više vremena, znanja i koncentracije nose više bodova. Prilikom rješavanja zadataka preporučuje se korištenje olovke i papira za skiciranje i razradu algoritma.



Primjer pravilno napisanog programa

Zadatak:

Napišite program koji će zbrojiti i oduzeti dva cijela broja.

Ulaz: U prvom retku se nalaze dva cijela broja A i B, međusobno odvojena jednim razmakom.

Izlaz: U prvi redak ispišite zbroj, a u drugi redak razliku brojeva A i B.

Rješenje u programskom jeziku C

```
#include <stdio.h>

int main()
{
    int a, b;
    scanf("%d%d", &a, &b);
    printf("%d\n", a+b);
    printf("%d\n", a-b);
    return 0;
}
```

Rješenje u programskom jeziku C++

```
#include <iostream>
using namespace std;

int main()
{
    int a, b;
    cin >> a >> b;
    cout << a+b << endl;
    cout << a-b << endl;
}
```

Rješenje u programskom jeziku Pascal

```
program p(input,output);
var
    a,b : integer;
begin
    read(a,b);
    writeln(a+b);
    writeln(a-b);
end.
```



Zadaci

U tablici možete pogledati ograničenja za zadatke:

Zadatak	Kvadrat	Palindrom	Stabla	DNA
Naziv izvornog kôda	kvadrat.cpp kvadrat.c kvadrat.pas	palindrom.cpp palindrom.c palindrom.pas	stabla.cpp stabla.c stabla.pas	dna.cpp dna.c dna.pas
Ulazni podaci	Standardni ulaz	Standardni ulaz	Standardni ulaz	Standardni ulaz
Izlazni podaci	Standardni izlaz	Standardni izlaz	Standardni izlaz	Standardni izlaz
Vremensko ograničenje	1 sekunda	1 sekunda	1 sekunda	0,2 sekunde
Memorijsko ograničenje	32 MB	32 MB	32 MB	32 MB
Broj bodova	20	40	60	80
Ukupno bodova	200			

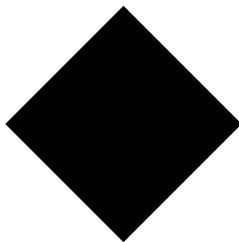


Zadatak: Kvadrat

Vrem. ograničenje: 1 sekunda / 30 bodova / Mem. ograničenje: 32 MB

Marica voli crtati geometrijske likove na računalu. Međutim, dosadili su joj svi programi za crtanje, pa je odlučila koristiti programe za obradu teksta.

Odlučila je uz pomoć znakova '*' nacrtati kvadrat, koji je zarotiran za 45 stupnjeva, kao na slici:



S obzirom da joj je cijeli svijet programa za obradu teksta nepoznat, potrebna joj je pomoć. Napišite program koji će Marici „iscrtati“ zadani kvadrat u tekstualnom obliku. Marica će zadati broj N , koji označava duljinu dijagonale kvadrata. Vaš zadatak je iscrtati zadani kvadrat uz pomoć znakova '*'.

Ulaz:

U prvom retku se nalazi **neparni** broj N ($3 \leq a \leq 99$) koji označava duljinu dijagonale kvadrata.

Izlaz:

U prvih N redaka je potrebno ispisati **po N znakova** koji će ocrtavati zadani kvadrat. Prazan prostor iscrtajte razmacima ' ', a područje kvadrata iscrtajte zvjezdicama '*'.

Test podaci:

	Test 1	Test 2	Test 3
Ulaz	3	5	7
Izlaz	<pre> * *** *</pre>	<pre> * *** ***** *** *</pre>	<pre> * *** ***** ***** ***** *** *</pre>
Napomene			



Zadatak: Palindrom

Vrem. ograničenje: 1 sekunda / 30 bodova / Mem. ograničenje: 32 MB

Darko i Jure su prije nekoliko dana naučili da je palindrom riječ koja se **čita jednako s desne i s lijeve strane**. S obzirom da se obojica vole nadmetati, osmislili su natjecanje u smišljanju palindroma, no to im je ubrzo dosadilo.

Sada su osmislili novo natjecanje u tome tko će napraviti najmanji mogući palindrom koji sadrži neku zadanu riječ. Jure nikako ne može podnijeti da ga Darko zove „noob“-om kad izgubi u igri, pa vas je zamolio da mu napišete program koji će ispisati **najkraći palindrom koji sadrži zadanu riječ**, tako da napokon može pobijediti Darka.

Ulaz:

U prvom i jedinom retku nalazi se jedna riječ koja se sastoji isključivo od malih slova engleske abecede. Riječ će imati **N** slova ($1 \leq N \leq 30$).

Izlaz:

U prvom i jedinom retku potrebno je ispisati najkraći palindrom koji sadrži zadanu riječ. Ako postoji više takvih palindroma potrebno je ispisati onaj koji je prvi po abecedi.

Test podaci:

	Test 1	Test 2	Test 3
Ulaz	abb	asd	udovicabacivo
Izlaz	abba	asdsa	udovicabacivodu
Napomene			



Zadatak: Stabla

Vrem. ograničenje: 1 sekunda / 30 bodova / Mem. ograničenje: 32 MB

Gradska uprava jednog malog grada odlučila je izgraditi ugostiteljski objekt u obližnjoj park-šumi. S obzirom da u park-šumi ima popriličan broj stabala, nastaju dva problema: gdje izgraditi novi ugostiteljski objekt i kolike površine treba biti.

Da ne bi nepotrebno sjekli stabla, i da bi udovoljili narodu, poručili su svim stanovnicima grada da se izjasne po tom pitanju. Stanovnici moraju na papir zapisati gdje bi oni smjestili ugostiteljski objekt i koliki bi površine bio taj ugostiteljski objekt.

Park-šuma je **pravokutnog oblika** i podijeljena je na **jedinične kvadrate**. Na svakom od kvadrata se može, a i ne mora nalaziti jedno stablo. Ugostiteljski objekt je također pravokutnog oblika, rubova paralelnih s rubovima park šume.

Stanovnici šalju prijedloge s koordinatama i dimenzijama ugostiteljskog objekta, i to u zapisu X, Y, H i W . Gdje su X i Y koordinate gornjeg lijevog ugla objekta, a W i H širina i visina objekta.

Kako je popriličan broj stanovnika poslao prijedloge, gradska vlast vas je zamolila za pomoć. Za svaki od prijedloga trebate izračunati **broj stabala** koji se **nalazi unutar** pojedinog **pravokutnika**, tj. predložene parcele za gradnju ugostiteljskog objekta. Gradska vlast uzet će na razmatranje samo one prijedloge koji zahtjevaju **minimalnu sječu stabala**. Nakon toga, gradska vlast može odlučiti gdje izgraditi ugostiteljski objekt, ali tako da posijeku što manje stabala.

Ulaz:

U prvom retku nalaze se dva prirodna broja R i S ($1 \leq R, S \leq 1000$), visina i širina park šume.

U sljedećih R redaka se nalazi po S znakova. Znak 'S' označava da se na tom mjestu nalazi stablo, a znak '.' (točka) označava da se na tom mjestu nalazi prazno mjesto.

U sljedećem retku nalazi se jedan prirodni broj N ($1 \leq N \leq 100,000$), koji predstavlja broj prijedloga stanovnika.

U sljedećih N redaka nalaze se po četiri prirodna broja X, Y, H i W koji predstavljaju koordinate i-tog predloženog ugostiteljskog objekta i njegove dimenzije.

Izlaz:

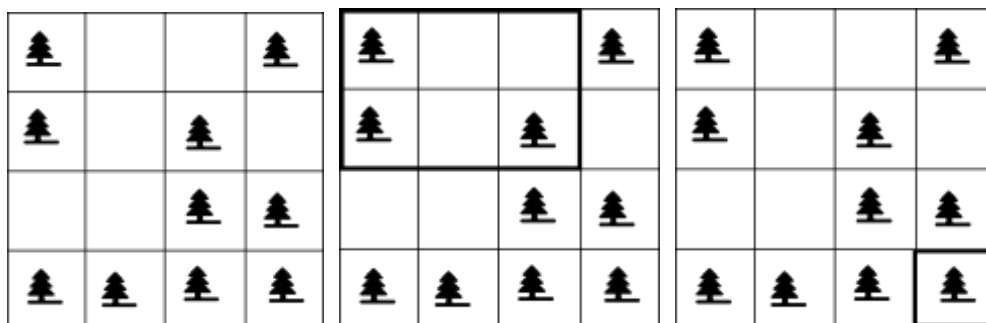
Ispisati broj prijedloga koji zahtjevaju minimalnu sječu stabala.



Test podaci:

	Test 1	Test 2	Test 3
Ulaz	3 3 .S. 1 1 1 3 3	4 4 S..S S.S. ..SS SSSS 2 1 1 2 3 4 4 1 1	4 2 S. S. 2 2 1 2 2 1 1 1 1
Izlaz	1	1	2
Napomene	(1)		

- (1) **Objašnjenje drugog test primjera:** Prva slika je sama park šuma, a druge dvije predstavljaju dva predložena pravokutnika za izgradnju ugostiteljskog objekta. Prvi prijedlog zahtjeva sječu tri stabla, dok drugi zahtjeva sječu samo jednog stabla. Dakle postoji samo jedan prijedlog koji ima minimalan broj posječenih stabala (samo jedno stablo).





Zadatak: DNA

Vrem. ograničenje: 1 sekunda / 30 bodova / Mem. ograničenje: 32 MB

Nekoliko znanstvenika koji se bave genetikom na jednom poznatom svjetskom institutu imaju problema s količinom podataka koje trebaju pohraniti. Naime, radi istraživanja u svoj sustav pohranjuju genetske kodove (nizove dušikovih baza) velikog broja ljudi. Svaki kod se sastoji isključivo od A, C, T i G slova.

Želeći izvršiti kompresiju kodova koje proučavaju, odlučili su određene kraće kodove dušikovih baza **zamijeniti kraticama**. Takve kodove nazivaju specijalnim kodovima. Pokušali su ručno zamjenjivati dijelove DNA koda kraticama, ali su shvatili da je to poprilično komplicirano. Zato su vas zamolili da im napišete program koji će izvršiti **optimalnu kompresiju** DNA koda koristeći njihove specijalne kodove i pripadajuće im kratice.

Znanstvenici će vam dati niz dušikovih baza jedne osobe, a potom i listu manjih specijalnih kodova za koje su dogovorili dužine zamjenskih kratica. Zanima ih s koliko se **najmanje** slova može zapisati zadani DNA kod, ako se kod optimalno komprimira koristeći njihove kratice.

Ulaz:

U prvom retku se nalazi DNA kod koji treba komprimirati, dužine **N** ($1 \leq N \leq 1\,000\,000$).

U drugom redu se nalazi broj **M** ($1 \leq M \leq 20$), koji predstavlja broj kratica.

U slijedećih **M** redaka se nalazi DNA niz i razmakom odvojen jedan broj koji predstavljaju i-ti DNA niz, maksimalne dužine **K** ($1 \leq K \leq 1000$) i dužinu kratice kojom se može zamijeniti **L** ($1 \leq L \leq 1000$).

Izlaz:

U prvom i jedinom retku potrebno je ispisati dužinu optimalno komprimiranog DNA koda.



Test podaci:

	Test 1	Test 2	Test 3
Ulaz	ATTCGAT 2 ATT 1 CGAT 2	TATACATAT 2 TACA 1 AT 1	GAGATAGACTAC 3 GA 1 AGA 2 AGAC 3
Izlaz	3	5	9
Napomene	(1)		

- (1) **Objašnjenje trećeg test primjera:** Optimalnu kompresiju dobijemo ako na ovaj način zamijenimo specijalne kodove (GA)(GA)T(AGAC)TAC. Umjesto dva slova „GA“ biti će kratica dužine jednog slova, umjesto „AGAC“ biti će kratica dužine 3 slova pa tako ukupna dužina ovog komprimiranog DNA niza je 9 slova (1 + 1 + T + 3 + TAC).