



5. ožujka 2012. od 09:00 do 12:00

Infokup 2012

Županijsko natjecanje / Srednja škola Algoritmi
(Pascal/C/C++)

Sadržaj

Upute za natjecatelje	1
Primjer pravilno napisanog programa	2
Zadaci.....	3
Zadatak: M	4
Zadatak: Parlament	5
Zadatak: Klingonci	6
Zadatak: Pustinja	7

Sponzori



Medijski pokrovitelji



Agencija za odgoj i obrazovanje
Education and Teacher Training Agency



MINISTARSTVO ZNANOSTI, OBRAZOVANJA
I ŠPORTA REPUBLIKE HRVATSKE

udruga mladih programera
dump



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	M	Parlament	Klingonci	Pustinja
Naziv izvornog kôda	m.cpp m.c m.pas	parlament.cpp parlament.c parlament.pas	klingonci.cpp klingonci.c klingonci.pas	pustinja.cpp pustinja.c pustinja.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	1 sekunda
Memorijsko ograničenje	32 MB	32 MB	32 MB	32 MB
Broj bodova	30	40	60	70
Ukupno bodova	200			

Sretno i uspješno!



Zadatak: M

30 bodova

M je ravnatelj britanske tajne službe MI6. Jedan od njegovih zadataka je i pridruživanje tajnih oznaka agentima ove službe. Oznaka agenta se sastoji od 4 znaka. Prvi znak je uvijek veliko slovo engleske abecede, a sljedeća tri su znamenke. Agentima se oznake pridružuju redom počevši od oznake A000 pa sve do oznake Z999 (A000-A999, B000-B999,...,Z000-Z999). Međutim, neke od tih oznaka se više ne koriste te ih M mora preskočiti i ne dodijeljivati. Naime, te oznake su umirovljene jer su ih u davna vremena nosili neki od najboljih agenata ove službe (npr. oznaku B007 je nosio James Bond).

Zbog nekoliko aktualnih svjetskih kriza, M je bio primoran zaposliti nove agente i svakom pridružiti svoju oznaku. Napiši program koji će ispisivati broj novo zaposlenih agenata ako su poznate oznake pridružene prvom i zadnjem novo zaposlenom agentu te sve umirovljene oznake.

Ulaz

- oznaka prvog novo zaposlenog agenta u obliku opisanom u zadatku;
- oznaka zadnjeg novo zaposlenog agenta u obliku opisanom u zadatku;
- prirodan broj N ($0 \leq N \leq 100$), broj umirovljenih oznaka;
- niz od N oznaka koje su umirovljene u slučajnom poretku

Izlaz

- prirodan broj koji predstavlja broj novo zaposlenih agenata.
-

Test podaci

	Test 1	Test 2	Test 3
Ulaz	B001 B010 1 A100	C995 D101 3 A101 D002 E050	A055 D055 5 A056 A057 A092 B100 E001
Izlaz	10	106	2997



Zadatak: Parlament

40 bodova

Ivica je zastupnik u Parlamentu. On i njegove kolege zastupnici svaki dan dolaze na posao. Za svakog zastupnika poznato je kad dolazi u parlament te koliko dugo tamo ostaje. Svakog dana svi zastupnici napuste parlament prije ponoći. Ivica je saznao da će sutra televizija prenositi zasjedanje. Televizijski će prijenos trajati **T** minuta, a on smije birati vrijeme početka te minute u kojima će se puštati reklame. Ukupno trajanje reklama može biti do **maksimalno R** minuta, ali se **ne smiju pustiti u prvoj i u zadnjoj** minuti TV prijenosa. Ivica želi odabrati takvo vrijeme početka prijenosa i takav raspored reklama da se ni u jednom trenutku **ne može vidjeti manje od Z** prisutnih zastupnika. Ako postoji više takvih rasporeda, Ivica bira onog s **najmanjim** ukupnim trajanjem reklama. Ako i dalje postoji više takvih rasporeda, Ivica bira **najranije** vrijeme početka prijenosa. Napišite program koji će pomoći Ivici!

Ulaz

Ivica će upisati: N, T, R, Z

Zatim N puta: H, M i O

N – broj zastupnika u parlamentu ($1 \leq N \leq 200$)

T – trajanje televizijskog prijenosa, u minutama ($1 \leq T \leq 500$)

R – najdulje dozvoljeno ukupno trajanje reklama, u minutama ($0 \leq R \leq T-2$)

Z – najmanji broj zastupnika koje želimo u svakom trenutku vidjeti na televiziji ($1 \leq Z \leq N$)

H i M – vrijeme dolaska n-tog zastupnika, sat i minuta ($0 \leq H \leq 23, 0 \leq M \leq 59$)

O – vrijeme zadržavanja n-tog zastupnika u parlamentu, u minutama ($1 \leq O \leq 500$)

Napomena: Ulazni podaci bit će takvi da će rješenje uvijek postojati.

Izlaz

Vaš program treba ispisati dva cijela broja odvojena razmakom koji opisuju odabrano vrijeme početka televizijskog prijenosa. Prvi broj je sat početka prijenosa (0-23), a drugi broj je minuta početka prijenosa (0-59).

Test podaci

	Test 1	Test 2	Test 3
Ulaz	1 60 2 1 10 30 100	3 100 10 2 8 15 200 9 12 190 9 20 70	3 30 15 2 8 50 10 8 55 90 9 10 15
Izlaz	10 30	9 12	8 55

Pojašnjenje trećeg test primjera: Kako nije moguće imati 2 zastupnika u parlamentu tijekom 30 minuta, prijenos počinje točno u 8 sati i 55 minuta. Nakon pet minuta prvi zastupnik odlazi iz parlamenta pa puštamo 10 minuta reklama dok ne dođe treći zastupnik.



Zadatak: Klingonci

60 bodova

Vjerojatno ste primijetili da je otkada Seve ima zaručnika... u domaćoj glazbi nemoguće naći mir. Srećom nedavno nas je posjetila klingonska opera. Svekoliki puk je bio očaran zavodljivim arijama klingonskih *slavuja*, pa su oni odlučili ostati u našoj zemlji. Odabrali bi neki hotel. Klingonci su poznati i po tome da zbog svoje ratničke naravi imaju vrlo lagan san, te ih i najmanji zvuk probudi. Međutim, manje je poznato da ti klingonski *slavuji*, čim im padne san na oči, toliko počnu piliti... kao jedna respektabilna međuzvezdana pilana. I zbog toga ne mogu spavati jedan blizu drugoga.

Ulaz

U prvom retku nalazi se brojevi **A**, **B** i **C**. $1 \leq A, B \leq 8$. $0 \leq C \leq A+B$. Prva dva broja označavaju veličinu hotela. Hotel je pravokutnog oblika s po jednim redom soba na svakom katu, a ti brojevi označavaju broj katova i broj soba na svakom katu. Treći broj označava broj zidova (bilo bočnih, bilo između katova) kroz koje klingonsko hrkanje može probiti (dijagonalno širenje zvuka se računa kao da je probilo i bočni i vertikalni zid).

Izlaz

U prvi i jedini redak treba ispisati koliko najviše klingonaca može spavati u hotelu (a da niti jedan ne čuje hrkanje drugog klingonca).

Test podaci

	Test 1	Test 2	Test 3																																																																																								
Ulaz	5 5 2	4 7 3	5 7 1																																																																																								
Izlaz	6 Pojašnjenje (jedan mogući raspored): <table><tr><td>K</td><td></td><td></td><td></td><td>K</td></tr><tr><td></td><td></td><td>K</td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td>K</td><td></td><td></td><td></td><td>K</td></tr><tr><td></td><td></td><td>K</td><td></td><td></td></tr></table>	K				K			K								K				K			K			5 Pojašnjenje (jedan mogući raspored): <table><tr><td></td><td>K</td><td></td><td></td><td></td><td>K</td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td>K</td><td></td><td></td><td></td></tr><tr><td>K</td><td></td><td></td><td></td><td></td><td></td><td>K</td></tr></table>		K				K												K				K						K	18 Pojašnjenje (jedan mogući raspored): <table><tr><td>K</td><td></td><td>K</td><td></td><td>K</td><td></td><td>K</td></tr><tr><td></td><td>K</td><td></td><td>K</td><td></td><td>K</td><td></td></tr><tr><td>K</td><td></td><td>K</td><td></td><td>K</td><td></td><td>K</td></tr><tr><td></td><td>K</td><td></td><td>K</td><td></td><td>K</td><td></td></tr><tr><td>K</td><td></td><td>K</td><td></td><td>K</td><td></td><td>K</td></tr></table>	K		K		K		K		K		K		K		K		K		K		K		K		K		K		K		K		K		K
K				K																																																																																							
		K																																																																																									
K				K																																																																																							
		K																																																																																									
	K				K																																																																																						
			K																																																																																								
K						K																																																																																					
K		K		K		K																																																																																					
	K		K		K																																																																																						
K		K		K		K																																																																																					
	K		K		K																																																																																						
K		K		K		K																																																																																					



Zadatak: Pustinja

70 bodova

Mirko je iznajmio slona preko interneta, pa putuje Saharom na svome slonu. On se svaki dan želi pomaknuti za jedno polje (gore, dolje, lijevo ili desno). Mirko zna da se njegov slon, kao i svaki drugi slon, izuzetno boji miševa, pa Mirko nikako ne želi proći poljem gdje ih ima, jer bi to moglo kritično završiti.

Mirko je slona iznajmio na neviđeno, pa mu nisu rekli da je slon malo čudan. Naime, slon **prvih T polja** sve Mirkove želje ispunjava, ali nakon toga uđe u nekakav neposlušni dio koji traje sve **dok slon ne pređe P narednih polja**. U neposlušnom dijelu, slon se također kreće u smjeru u kojem Mirko želi, međutim, na **točno jednom** od tih P polja, slon će **jednom stati** (a Mirko ne zna kada) i **točno se P dana neće pomicati** s mjesta, pa makar i bio na kranjem mjestu gdje Mirko želi stići. Nakon što je prešao tih P polja, ponovo će T polja biti poslušan, pa će sljedećih P polja biti neposlušan itd.

Pustinjska klima nije ni malo ugodna. Pustinjom se izmjenjuju sunce i hlad. Na suncu je jako vruće, a u hladu jako hladno. Mirko kada je kretao na put, nije se baš najbolje pripremio, pa uzastopno može izdržati najviše maxT dana na suncu, ili uzastopno najviše **MaxT** dana u hladu.

Kada je iznajmljivao slona, Mirku su dali i kartu pustinje na kojoj je svako polje označeno jednim od tri znaka:

- '#' – na polju ima miševa, pa je polje neprohodno
- '.' – prohodni dio pustinje koji se nalazi na suncu
- '\$' – prohodni dio pustinje koji se nalazi u hladu

Mirko želi doći na željeno mjesto u pustinji, ali nikako ne želi biti u opasnosti da strada ako ima miševa ili ako bi spletom okolnosti mogao previše uzastopnih dana provesti na suncu ili u hladu.

Ako su poznate Mirkova početna pozicija i mjesto na koje želi doći, napišite program koji računa duljinu najkraćeg puta, na kojem neće biti opasnosti. Ukoliko takav put ne postoji, program treba vratiti -1.

Ulaz

U prvom retku nalaze dva broja **N, M** ($1 \leq N, M \leq 50$), visina i širina pustinje.

U sljedećem retku su tri broja **T, P, MaxT** ($1 \leq T, P, \text{MaxT} \leq 20$).

U trećem i četvrtom retku nalaze se po dva broja (**N1, M1**) i (**N2, M2**), ($0 \leq N1, N2 < N$ i $0 \leq M1, M2 < M$) koordinate Mirkove početne pozicije i koordinate mjesta na koje želi doći.

U sljedećih N redaka se nalazi po M znakova koji predstavljaju svako polje pustinje.

Napomena: Mirkova početna pozicija i mjesto na koje želi stići neće nikada biti na istom polju ni na neprohodnom polju. Koordinata (0,0) nalazi se u gornjem lijevom kutu.

Ukoliko npr. Mirko kreće s pozicije (0, 0) i treba doći na poziciju (1,0), a na toj poziciji sija sunce, neovisno o tome je li na poziciji (0, 0) bilo sunce ili hlad, duljina puta je 1, i on je jedan dan proveo na suncu.

Izlaz

U prvi i jedini redak treba ispisati duljinu najkraćeg puta na kojem ne vrebaju opasnosti.



Test podaci

	Test 1	Test 2	Test 3
Ulaz	3 5 4 3 6 0 0 2 3 ...\$. .###.\$	7 4 2 3 6 0 0 6 2\$.\$.\$.	6 6 1 3 5 0 0 5 0 ...\$. \$. ...\$.\$.
Izlaz	7	8	-1

Pojašnjenje drugog test primjera:

Mirko može ići preko sljedećih polja:

(0,0) -> (0,1) -> (0,2) -> (1,2) -> (2,2) -> (3,2) -> (4,2) -> (5,2) -> (6,2)

Tijekom putovanja proći će kroz dva neposlušna dijela i dvaput stajati.

U zgradama su koraci unutar neposlušnog dijela: 1, 2, 3, [4, 5, 6], 7, 8, [9, 10, 11], 12, ...