

Sveučilište u Zagrebu  
**Fakultet Elektrotehnike i Računarstva**

Natjecateljsko Programiranje  
**PRIJEMNI ISPIT**

**5.9.2009.**

Poštovane studentice i studenti FER-a!

Ispit (natjecanje) se sastoji od nekoliko zadataka različite težine. Zadaci **nisu** poredani po težini. Svi zadaci nose **jednak** broj bodova. Unatoč tome što je vrijeme rješavanja 24 sata, od vas se ne očekuje da ćete svo vrijeme rješavati zadatke. Težina zadataka je takva da većina natjecatelja neće uspjeti riješiti sve zadatke, tako da ako i ne riješite neki zadatak, nemojte biti obeshrabreni nego rješavajte ostale zadatke.

Dobro pročitajte poglavlja ULAZNI PODACI i IZLAZNI PODACI kod svakog pojedinog zadatka. Na tom mjestu piše sve o formatu ulaznih i izlaznih podataka koji mora biti **strogo poštovan** (tj. **ne smijete** ispisivati nikakav dodatni tekst, kao "Rješenje je" i slično) kako bi vaša rješenja bila točno evaluirana. Za ilustraciju i bolje razumijevanje pogledajte poglavlje TEST PRIMJERI u kojem će se nalaziti nekoliko jednostavnijih primjera ulaznih podataka i njima odgovarajućih izlaznih podataka.

**Također, vaši programi ne smiju pristupati nikakvim datotekama niti ih kreirati. Kršenje ovog pravila rezultirat će gubitkom bodova za taj zadatak.**

Obratite pažnju da osim vremenskog ograničenja postoji i memorijsko ograničenje. Ako će vaš program prilikom izvršavanja koristiti više memorije nego što je dozvoljeno, neće dobiti bodove za taj zadatak.

**Sustav za evaluaciju nalazi se na računalu s LINUX operativnim sustavom. Stoga, da bi se vaš program uspješno evaluirao, ne smijete koristiti naredbe i funkcije specifične za DOS. Tako npr. u Pascalu ne smijete koristiti CRT modul, a u C/C++-u conio.h i slične module.**

**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 će generirati točan rezultat i regularno završiti svoje izvođenje unutar predviđenog vremenskog i memorijskog ograničenja.

Program u Pascal-u treba se izvršiti do kraja tj. do 'end.' ili ako za završetak izvršavanja koristite naredbu 'halt', treba je koristiti bez parametara ili kao 'halt(0)'. Program u C/C++-u treba se izvršiti do kraja tj. do 'return 0;' u funkciji 'main' koja treba biti deklarirana kao 'int main(void)' ili ako za završetak izvršavanja koristite naredbu 'exit', treba je koristiti kao 'exit(0)'.

Da bi program koji rješava problem iz nekog zadatka dobio maksimalni broj bodova, primijenjeni algoritam mora biti valjan i efikasan tj. brz. Test podaci su unaprijed osmišljeni i koncipirani na način da će programi koji koriste neke manje efikasne, ali valjane algoritme, također dobiti određeni broj bodova (npr. od ukupno 60 bodova, jako loš i spor algoritam će dobiti npr. 20 bodova, dok će dobar algoritam, ali ne i najbolji dobiti npr. 40 bodova). Programi koji će raditi jako brzo za sve test podatke, ali neće davati točne rezultate, naravno, neće donositi bodove. Znači, valjanost algoritma je na prvom mjestu, a brzina izvršavanja na drugom.

Prilikom rješavanja zadataka obratite pažnju na **ograničenja** kod ulaznih podataka jer su ona izrazito bitna za osmišljavanje efikasnog algoritma.

Kod svakog zadatka naći ćete nekoliko test primjera koji vam mogu pomoći da bolje shvatite zadatak. Međutim, ako vaš program radi ispravno za zadane test primjere to još uvijek **nije garancija** niti da vaš program radi korektno za ostale test podatke niti da je dovoljno brz.

**Natjecateljsko Programiranje - Prijemni ispit**

**Vremensko ograničenje: 5 sec  
Memorijsko ograničenje: 256 MB**

# ZBROJ

---

Zadana su dva broja, **A** i **B**, manja ili jednaka 10. Kako je ovo zadatak predviđen za upoznavanje sa sustavom od Vas se ne zahtijeva ništa više nego da ih zbrojite.

## Ulazni podaci

U prvom i jedinom retku ulaza nalaze se dva prirodna broja, **A** i **B** ( $A, B \leq 10$ ) međusobno odvojena razmakom.

## Izlazni podaci

U prvi i jedini redak izlaza potrebno je ispisati zbroj brojeva **A** i **B**.

## Test primjeri

**ULAZ :**

2 2

**IZLAZ :**

4

**ULAZ :**

5 6

**IZLAZ :**

11

**ULAZ :**

1 1

**IZLAZ :**

2

# ROBINSON

---

William Heath Robinson bio je poznati britanski crtač stripova i ilustracija. Najpoznatiji je po svojim crtežima uređaja upitne korisnosti i ponekad fizikalnom stvarnošću neograničene konstrukcije. Kako bi pokazali da fizikalna stvarnost ne vrijedi više puno, od vas se očekuje da napravite program koji će simulirati uređaje koje je nacrtao gospodin Robinson. Za potrebe ovog zadatka jedan uređaj se sastoji od **točno jedne** kuglice, neograničenog broja vrlo glatkih metalnih blokova i **najviše deset** metalnih cijevi.

Uređaj je u digitalnom obliku prikazan kao pravokutna mreža sa  $R$  redaka i  $S$  stupaca. Kuglica je prikazana sa znakom '\*', metalni blokovi znakom '#', a otvori svake cijevi znakovima '0' do '9'. Znakom '.' prikazano je prazno polje.

U svakom trenutku simulacije, kuglica pada prema dolje, to jest pomiće se iz retka  $R$  u redak  $R + 1$ . Ako se u retku  $R + 1$  nalazi metalni blok tada se kuglica od istog odbija. Smjer odbijanja ovisi o poljima koji okružuju kuglicu:

- Ukoliko se lijevo od kuglice (u retku  $R$  i stupcu  $S - 1$ ) nalazi metalni blok kuglica se odbija u desno polje  $(R, S + 1)$ .
- Ukoliko se desno od kuglice (u retku  $R$  i stupcu  $S + 1$ ) nalazi metalni blok kuglica se odbija u lijevo polje  $(R, S - 1)$ .
- Ukoliko su oba polja prazna, ili zauzeta blokovima kuglica se zaustavlja na polju  $(R, S)$ .

Ako se u retku  $R + 1$  nalazi otvor cijevi tada kuglica prolazi kroz cijev i istog trena izlazi na drugom otvoru iste cijevi. Također, ako se kuglica odbije u otvor cijevi, ona istog trena izlazi na drugom otvoru iste cijevi.

Vaš zadatak je da za zadan uređaj i proteklo vrijeme odredite na kojem se mjestu nalazi kuglica.

## Ulazni podaci

U prvom retku nalaze se dva prirodna broja,  $R$  i  $S$ , oba manja od 100 i prirodni broj  $T$ , broj proteklih trenutaka, manji od 1000.

U sljedećih  $R$  redaka nalazi se  $S$  znakova '\*', '#', '.' i '0' do '9'.

U cijelom unosu nalaziti će se točno jedan znak '\*'.

Prvi i zadnji redak sadrže isključivo znakove '#'.

Prvi i zadnji znak u svakom retku je '#'.

Znakovi '0' do '9' pojavljuju se isključivo nula ili dva puta u cijeloj mreži.

## Izlazni podaci

U prvi i jedini redak ispišite u kojem retku i stupcu se nalazi kuglica nakon  $T$  proteklih trenutaka.

## Test primjeri

**ULAZ :**

5 5 2

#####

#..\*#

#...#

#...#

#####

**IZLAZ :**

3 3

**ULAZ :**

5 5 1

#####

#..\*#

#1.1#

#...#

#####

**IZLAZ :**

3 2

**ULAZ :**

5 5 2

#####

#..\*#

#1.1#

#...#

#####

**IZLAZ :**

4 2

### Pojašnjenje prvog test primjera:

Početno stanje	Nakon prvog trenutka	Nakon drugog trenutka
#####	#####	#####
#..*#	#.*.#	#...#
#...#	#...#	#.*##
#...#	#...#	#...#
#####	#####	#####

# UMNOŽAK

---

Zadan je niz od  $N$  nenegativnih cijelih brojeva. Smijete najviše  $K$  puta izvesti operaciju uvećavanja nekog člana niza za jedan. Svako izvođenje operacija je zasebno i može se izvoditi na **bilo kojem članu niza**.

Nađite optimalan izbor uvećavanja članova tako da **umnožak članova u nizu bude najveći** mogući.

## Ulazni podaci

U prvom redu nalaze se dva pozitivna broja  $N$  ( $N \leq 100\,000$ ) i  $K$  ( $K \leq 1\,000\,000\,000$ ). Oni redom označavaju duljinu niza, te broj uvećavanja koje je dopušteno napraviti.

U sljedećih  $N$  redaka nalazi se po jedan nenegativni cijeli broj,  $a_i$  ( $a_i \leq 1\,000\,000\,000$ ) koji predstavlja  $i$ -ti član niza.

## Izlazni podaci

U  $N$  redaka potrebno je ispisati  $N$  brojeva koji predstavljaju novi niz. Brojevi moraju biti ispisani istim redoslijedom kojim su navedeni u ulaznim podacima.

## Test primjeri

**ULAZ :**

3 2  
1  
5  
2

**ULAZ :**

1 1  
1

**ULAZ :**

5 100  
500  
200  
300  
400  
100

**IZLAZ :**

3  
5  
2

**IZLAZ :**

2

**IZLAZ :**

500  
200  
300  
400  
200

### Objašnjenje prvog test primjera:

Točna rješenja su (3, 5, 2) i (2, 5, 3). U oba slučaja najveći mogući umnožak je 30.

# ISPIT

---

Na Fakultetu Elektrotehnike i Računarstva nije neobično da se u istoj dvorani istovremeno pišu dva ispita iz raznorodnih predmeta, na primjer Operacijskih Sustava i Električnih Polja. U jednoj od mnogih takvih dvorana nalazi se  $N$  klupa. Svaka klupa je dovoljno velika za **samo dva** studenta.

Kako studenti polako dolaze na ispite, asistenti ih smještaju u klupe i to na sljedeći način:

Asistent prvo pita studenta kojem ispitu pristupa. Tada redom pregledava klupe i smješta studenta u praznu klupu ili klupu u kojoj se već nalazi jedan student koji **ne pristupa istom ispitu**. Ukoliko je više mogućih klupa asistent uvijek odabire onu s **najmanjim rednim brojem**.

Dvorana je **na početku prazna**.

Zadan je redosljed pristizanja studenata. Napišite program koji će odrediti koliko je ljudi smješteno u kojoj klupi nakon što svi studenti pristupe ispitu.

## Ulazni podaci

U prvom redu nalazi se broj  $N$  ( $1 \leq N \leq 100$ ), broj klupa i  $S$  ( $1 \leq S \leq 200$ ), broj studenata koji pristupaju ispitima.

U sljedećih  $S$  redaka nalazi se po jedan broj 1 ili 2, koji označava ispite kojima studenti pristupaju, redom kojim ulaze u dvoranu.

Broj studenata i ispiti kojima pristupaju će uvijek biti takvi da ih je moguće rasporediti.

## Izlazni podaci

Izlaz se sastoji od  $N$  redaka. U  $i$ -tom retku potrebno je ispisati broj studenata koji sjede u klupi s rednim brojem  $i$ .

## Test primjeri

**ULAZ :**

2 3  
1  
2  
1

**IZLAZ :**

2  
1

**ULAZ :**

3 3  
2  
2  
2

**IZLAZ :**

1  
1  
1

**ULAZ :**

2 2  
2  
1

**IZLAZ :**

2  
0

# UNIJA

---

Interval prirodnih brojeva zadan je prirodnim brojevima  $A$  i  $B$ , te sadrži sve prirodne brojeve između  $A$  i  $B$  (uključivo), a označavamo ga s  $[A, B]$ . Tako, na primjer, interval  $[2, 5]$  sadrži brojeve 2, 3, 4 i 5, a interval  $[100, 100]$  samo broj 100.

Uniju  $N$  intervala definiramo kao skup brojeva koji se nalaze u **barem jednom** od zadanih intervala. Na primjer, unija intervala  $[2, 4]$ ,  $[3, 5]$  i  $[8, 9]$  je skup  $\{2, 3, 4, 5, 8, 9\}$ .

Napišite program koji će ispisati **ukupan broj** prirodnih brojeva koji se nalaze u **uniji**  $N$  zadanih intervala.

## Ulazni podaci

U prvom redu nalazi se prirodan broj  $N$  ( $N \leq 100\,000$ ), broj intervala. U sljedećih  $N$  redaka nalaze se po dva prirodna broja  $A_i$  i  $B_i$  ( $1 \leq A_i \leq B_i \leq 1\,000\,000\,000$ ), granice  $i$ -tog intervala.

## Izlazni podaci

U prvi i jedini redak potrebno je ispisati ukupan broj prirodnih brojeva koji se nalaze u uniji intervala.

## Test primjeri

**ULAZ :**

3  
2 4  
3 5  
8 9

**IZLAZ :**

6

**ULAZ :**

2  
200000000 300000000  
100000000 200000000

**IZLAZ :**

200000001



# KOLEKCIJA

Mirko već godinama sakuplja plastične figurice, koje su mu već polako dojadile. Postoji **deset** vrsta figurica koje se razlikuju po bojama. Neki dan je otvorio oglase i pronašao **N** oglasa u kojima se traže takve plastične figurice, te je odlučio **prodati sve svoje figurice** kako bi povratio uloženi novac.

U svakom oglasu navedena je količina figurica svake boje koje autor oglasa traži. Autori oglasa pritom **ne pristaju na polovične ponude** (višak ili manjak), pa ukoliko Mirko ne zadovolji propisanu količinu figurica od svake boje, ne pristaju na ponudu. Tako na primjer autor koji zahtjeva 5 crvenih i 3 plave figurice ne pristaje na ponudu od 4 crvene i 3 plave figurice, kao niti primjerice na ponudu od 5 crvenih 3 plave i 2 zelene figurice.

Napišite program koji će pomoći Mirku da odgonetne na koje oglase se treba javiti kako bi prodao sve figurice koje ima, ako je to moguće. Rješenje **ne** mora nužno biti jedinstveno; u tom slučaju ispišite **bilo koje točno rješenje**.

## Ulazni podaci

U prvom redu nalazi se prirodni broj **N** ( $N \leq 20$ ), broj oglasa.

U sljedećih **N** redaka nalazi se opis pojedinog oglasa. Svaki red sadrži točno 10 nenegativnih cijelih brojeva **C<sub>i</sub>** ( $C_i \leq 10^7$ ), koji redom predstavljaju traženi broj figurica po bojama.

Posljednji redak ima isti format kao prethodnih **N** redaka i opisuje Mirkovu kolekciju. U tom redu bar jedan broj će biti veći od nule.

## Izlazni podaci

U prvom i jedinom retku potrebno je ispisati redne brojeve ponuda koje Mirko mora prihvatiti kako bi prodao svoju kolekciju.

Ukoliko rješenje ne postoji, u prvi i jedini redak ispišite 0.

## Test primjeri

**ULAZ :**

3

1 0 0 0 0 0 0 0 0 0

0 3 0 0 0 0 0 0 0 0

1 3 0 0 0 0 0 0 0 0

1 3 0 0 0 0 0 0 0 0

**IZLAZ :**

3

**ULAZ :**

3

1 0 0 0 0 0 0 0 0 0

0 3 0 0 0 0 0 0 0 0

1 3 1 0 0 0 0 0 0 0

1 3 0 0 0 0 0 0 0 0

**IZLAZ :**

2 1