

## zadaci

---

zadatak	fliper	memorija	rupe	swap
izvorni kôd	fliper.pas fliper.c fliper.cpp	memorija.pas memorija.c memorija.cpp	rupe.pas rupe.c rupe.cpp	swap.pas swap.c swap.cpp
ulazni podaci	standardni ulaz			
izlazni podaci	standardni izlaz			
vremensko ograničenje (pentium4 2.4ghz)	1 sec	1 sec	0.5 sec	0.5 sec
broj bodova	5	5	5	5
	20			

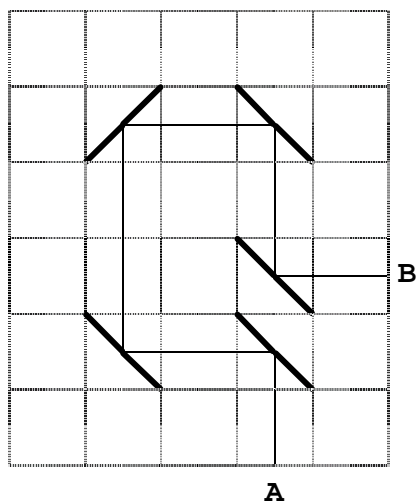
Mirko je za rođendan dobio jeftinu kopiju prvog originalnog drvenog flipera.

Ploča za fliper je pravokutnog oblika i sastoji se od **kvadratnih polja** poredanih u M redaka i N stupaca. Recimo na ploči označeni su brojevima od 1 do M odozgo prema dolje, a stupci su označeni brojevima od 1 do N slijeva na desno.

Na nekim poljima nalaze se **prepreke** tj. dijagonalne dužine koje spajaju ili gornji lijevi i donji desni kut tog polja (ovaj smjer označavamo **brojem 1**) ili donji lijevi i gornji desni kut tog polja (ovaj smjer označavamo **brojem 2**).

Loptica kreće iz početne rupe koja se nalazi **na rubu ploče** i to u centru nekog brida nekog rubnog **polja A** u smjeru centra tog polja i nastavlja kretanje po ploči odbijajući se od prepreka koje joj stoje na putu pod kutem od **90 stupnjeva**. Na kraju, loptica ulazi u završnu rupu koja se nalazi također **na rubu ploče** u centru nekog brida nekog rubnog **polja B**.

Na sljedećoj ilustraciji nalazi se 3. test primjer:



Jednog dana, dok je Mirko s razredom bio u Pionirskom gradu, mlađi brat ukrao mu je jednu prepreku s ploče i pojeo je.

Napišite program koji će **odrediti neko prazno polje ploče** na koje treba postaviti prepreku kako bi loptica mogla nakon što izađe iz početne rupe kretanjem doći u završnu rupu. Novu prepreku je potrebno postaviti čak i u slučaju da loptica ide početne do završne rupe i bez te nove prepreke.

Ako postoji više rješenja, potrebno je odrediti ono rješenje kod kojeg je **ukupni put** kojeg će loptica prijeći **što manji**.

### ulazni podaci

U prvom retku ploče nalaze se tri prirodna broja  $M$ ,  $N$  i  $K$ ,  $1 \leq M, N \leq 1000$ ,  $1 \leq K \leq 1,000,000$  (milijon), broj redaka i broj stupaca na ploči i ukupni broj prepreka (prije krađe jedne prepreke).

U drugom retku nalaze se tri prirodna broja,  $R_A$ ,  $S_A$  i  $T_A$ ,  $1 \leq R_A \leq M$ ,  $1 \leq S_A \leq N$ ,  $1 \leq T_A \leq 4$ . Brojevi  $R_A$  i  $S_A$  označavaju broj retka i broj stupca polja  $A$ , a broj  $T_A$  je oznaka brida na polju  $A$  s kojeg loptica kreće u igru.

U trećem retku nalaze se tri prirodna broja,  $R_B$ ,  $S_B$  i  $T_B$ ,  $1 \leq R_B \leq M$ ,  $1 \leq S_B \leq N$ ,  $1 \leq T_B \leq 4$ . Brojevi  $R_B$  i  $S_B$  označavaju broj retka i broj stupca polja  $B$ , a broj  $T_B$  je oznaka brida na polju  $B$  na kojem loptica završava.

Bridovi nekog polja označeni su brojevima 1 (gore), 2 (desno), 3 (dolje) ili 4 (lijevo).

U sljedećih  $K-1$  redaka nalaze se podaci o preprekama (bez ukradene prepreke). U svakom od tih redaka nalaze se tri prirodna broja  $R$ ,  $S$  i  $T$ ,  $1 \leq R \leq M$ ,  $1 \leq S \leq N$ ,  $1 \leq T \leq 2$ . Brojevi  $R$  i  $S$  označavaju broj retka i broj stupca polja na kojem se nalazi prepreka, a broj  $T$  označava smjer prepreke.

### izlazni podaci

U prvi i jedini redak ispišite tri prirodna broja  $R$ ,  $S$  i  $T$ . Brojevi  $R$  i  $S$  označavaju broj retka i broj stupca polja na koje moramo postaviti prepreku u smjeru  $T$ .

**Napomena:** test podaci će biti takvi da će rješenje (iako ne nužno jedinstveno) uvijek postojati.

### test primjeri

**ulaz**

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

**izlaz**

2 2 1

**ulaz**

5 5 9  
4 1 4  
2 5 2  
1 4 2  
2 3 1  
2 5 2  
4 1 2  
4 2 2  
4 3 2  
5 2 1  
5 5 2

**izlaz**

2 1 2

**ulaz**

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

**izlaz**

5 2 1

## memorija

---

Programski jezik za upravljanje memorijom sastoji se od nekoliko jednostavnih naredbi za zauzimanje i oslobađanje memorije.

Memorija kojom upravlja naš program sastoji se od  $N$  uzastopnih memorijskih lokacija redom označenih brojevima od 1 do  $N$ . Svaka lokacija može biti slobodna ili zauzeta, a na početku su sve lokacije slobodne.

U svakom retku programa nalazi se jedna od sljedećih naredbi:

- 1 A B** ova naredba **zauzima**  $B$  uzastopnih memorijskih lokacija počevši od lokacije  $A$ . Prije izvršavanja ove naredbe sve memorijske lokacije na koje se ova naredba odnosi bit će slobodne.
- 2 A B** ova naredba **oslobađa**  $B$  uzastopnih memorijskih lokacija počevši od lokacije  $A$ . Prije izvršavanja ove naredbe sve memorijske lokacije na koje se ova naredba odnosi bit će zauzete (mada nije nužno da su te memorijske lokacije prethodno bile zauzete jednom naredbom).
- 3** ova naredba **ispisuje** duljinu najduljeg niza uzastopnih slobodnih memorijskih lokacija.

Napišite program koji generira izlaz zadanog programa.

### ulazni podaci

U prvom retku nalaze se dva prirodna broja  $N$  i  $K$ ,  $1 \leq N \leq 16,000$ ,  $3 \leq K \leq 200,000$ , ukupni broj memorijskih lokacija i broj naredbi u programu.

U svakom od sljedećih  $K$  redaka nalazi se jedna **legalna** naredba zadanog programskog jezika, redom izvršavanja. Barem jedna od tih naredbi bit će naredba za ispis.

### izlazni podaci

Vaš program treba ispisati sve što ispisuju naredbe za ispis tijekom izvođenja zadanog programa.

### test primjeri

#### ulaz

5 3  
1 3 2  
3  
3

#### izlaz

2  
2

#### ulaz

7 4  
1 6 1  
3  
1 4 1  
3

#### izlaz

5  
3

#### ulaz

10 7  
1 1 5  
1 6 5  
3  
2 2 3  
3  
2 5 3  
3

#### izlaz

0  
3  
6

Mirko već danih neprestano igra jednu jako zanimljivu igru i pritom se ludo zabavlja.

Ploča za igru je pravokutnog oblika i sastoji se od polja poredanih u 2 retka i N stupaca.

Na nekim poljima se nalaze žetoni, a na svakom od njih je ispisana vrijednost tog žetona tj. broj različit od nule, veći od ili jednak -10 i manji od ili jednak 10.

**Vrijednost ploče** računamo tako da u svim stupcima u kojima se nalaze dva žetona pomnožimo njihove vrijednosti, a sve dobivene umnoške zbrojimo.

-3	-1	-2		5	-1	
	-3	2	4		5	-2

Tako npr. vrijednost gore prikazane ploče (iz 3. test primjera) iznosi  $(-1 \cdot -3) + (-2 \cdot 2) + (-1 \cdot 5) = -6$ .

Međutim, Mirko može povećati vrijednost ploče pomicanjem žetona unutar jednog retka. Preciznije, on **smije pomaknuti** svaki žeton na neko od slobodnih polja lijevo ili desno (**za jedno ili više polja**), ali pri tome **ne smije preskakati** ostale žetone u tom retku.

-3	-1	-2			5	-1
-3			2	4	5	-2

Vrijednost gore prikazane ploče sada je  $(-3 \cdot -3) + (5 \cdot 5) + (-1 \cdot -2) = 36$ .

Napišite program koji će odrediti kolika je **maksimalna vrijednost ploče** koju Mirko može dobiti pomicanjem žetona prema navedenim pravilima.

### ulazni podaci

U prvom retku nalazi se prirodni broj N,  $1 \leq N \leq 400$ , broj stupaca na ploči.

U svakom od sljedeća dva retka nalazi se po N cijelih brojeva. Ti brojevi opisuju izgled ploče (tj. vrijednosti svih žetona na ploči) na početku igre. Polja ploče na kojima se ne nalazi žeton označena su brojem 0.

### izlazni podaci

U prvi i jedini redak ispišite traženu maksimalnu vrijednost ploče.

### test primjeri

**ulaz**

2  
3 0  
0 2

**izlaz**

6

**ulaz**

3  
-1 -1 -2  
-1 0 2

**izlaz**

-1

**ulaz**

7  
-3 -1 -2 0 5 -1 0  
0 -3 2 4 0 5 -2

**izlaz**

36

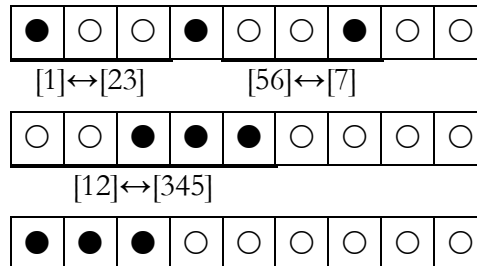
## swap

Ploča za igru sastoji se od **N polja** koja su redom slijeva na desno označena brojevima od 1 do N.

Na svakom polju ploče nalazi se crna ili bijela kuglica, a niz od jedne ili više uzastopnih kuglica iste boje okruženih s kuglicama druge boje ili krajevima ploče nazivamo **grupa**.

Jedna **zamjena** sastoji se od međusobne zamjene dvije **susjedne** grupe. U jednom **koraku** smijemo izvršiti **najviše K** simultanih zamjena, a svaka kuglica u nekom koraku može sudjelovati samo u jednoj zamjeni.

Na sljedećoj ilustraciji nalazi se 3. test primjer:



**Cilj igre** je pomoću gore definiranih zamjena **premjestiti sve kuglice iste boje u jednu grupu**.

Napišite program koji će odrediti neki niz zamjena pri kojem će **broj koraka** biti **što manji**.

### ulazni podaci

U prvom retku nalaze se dva prirodna broja N i K,  $1 \leq N \leq 100,000$  (sto tisuća),  $1 \leq K \leq 10,000$  (deset tisuća).

U drugom retku nalazi se N brojeva međusobno odvojenih s po jednim razmakom. Svaki od tih brojeva označava jednu kuglicu na ploči (redom slijeva na desno) - broj 0 označava crnu kuglicu, a broj 1 bijelu.

### izlazni podaci

Svaki korak treba ispisati u svoj redak, redom izvršavanja.

Prvi broj unutar jednog retka je prirodni broj Z tj. broj simultanih zamjena u tom koraku. Nakon njega slijedi Z brojeva. Svaki od tih brojeva označava broj polja na kojem se nalazi najlijevija kuglica unutar te zamjene.

U zadnji redak ispišite broj 0.

**Napomena:** rješenje ne mora biti jedinstveno.

### test primjeri

**ulaz**

3 1  
0 1 1

**izlaz**

0

**ulaz**

5 2  
1 0 1 0 1

**izlaz**

2 1 3  
1 2  
0

**ulaz**

9 3  
0 1 1 0 1 1 0 1 1

**izlaz**

2 1 5  
1 1  
0