



**DANI MLADIH INFORMATIČARA HRVATSKE 2009**  
Dubrovnik, 22.-26. ožujka

**SREDNJOŠKOLSKA SKUPINA, PASCAL/C/C++**  
**I. podskupina, Drugi dan natjecanja**

---

| <b>ZADATAK</b>                | <b>KLONOVI</b>   | <b>BAGER</b> | <b>PČELICE</b> |
|-------------------------------|------------------|--------------|----------------|
| <b>ulazni podaci</b>          | standardni ulaz  |              |                |
| <b>izlazni podaci</b>         | standardni izlaz |              |                |
| <b>vremensko ograničenje</b>  | 1 sekunda        | 2 sekunde    | 2 sekunde      |
| <b>memorijsko ograničenje</b> | 64 MB            |              |                |
| <b>broj bodova</b>            | <b>50</b>        | <b>70</b>    | <b>80</b>      |
|                               | <b>200</b>       |              |                |



Simona se bavi uzgojem ovaca. Kako joj se ne da čekati da se stado poveća prirodnim putem, izgradila je postrojenje za kloniranje koje se sastoji od  $N$  strojeva.

Strojevi za kloniranje su fascinantna dostignuća tehnologije. Oni na sebi imaju tipkovnicu na kojoj možemo utipkati neki prirodni broj  $p$ . Ako se u stroju nalazi  $K$  ovaca, pritiskom na „Enter“ počinje kloniranje, te u idućem trenu iz stroja bleji čak  $p \cdot K$  ovaca! Međutim, iz nekog razloga kojeg Simona ne može dokučiti, kloniranje radi samo ako je  $p$  **prost** broj. Također, svaki stroj ima maksimalni kapacitet – najveći broj ovaca koji može odjednom sadržavati.

Simoni je dosadilo prešetaivati se od stroja do stroja i pritiskati tipke. Zato je zaposlila  $N$  pomoćnika, jednog kraj svakog stroja, te sada urla naredbe pomoćnicima, šecući se pritom po dvorištu:

- UNESI  $p$  – svi pomoćnici poslušno utipkaju broj  $p$  na tipkovnicu; broj  $p$   **mora biti prost**.
- KLONIRAJ  $a_1 a_2 \dots a_K$  – pomoćnici kod strojeva  $a_1, a_2, \dots, a_K$  svi pritisnu „Enter“ i ovce u tim strojevima se kloniraju. Simona pazi da redni brojevi  $a_1, a_2, \dots, a_K$  svi budu međusobno različiti da se pomoćnici ne zbune. Kako Simona nema baš dobru koncentraciju, ona  **nikad neće navesti više od  $M$  rednih brojeva strojeva u jednoj naredbi KLONIRAJ**.

Na početku je Simona u svaki stroj stavila po  **jednu** ovcu. Kako bi u potpunosti iskoristila strojeve, Simona namjerava **svaki stroj** popuniti do njegovog  **maksimalnog kapaciteta**. Jednom kada niz naredbi krene, nije dozvoljeno vaditi ovce iz strojeva. Da joj grlo ne bi previše stradalo (jer ima još mnogo estradnih ambicija), ona želi izurlati  **što je moguće manje naredbi** tako da se svaki od strojeva  **potpuno napuni ovcama**.

Napišite program koji će ispisati neki  **najkraći** slijed naredbi koje Simona treba urlati.

### **ULAZNI PODACI**

U prvom redu nalazi se prirodni broj  $N$  ( $1 \leq N \leq 50$ ), broj strojeva za kloniranje.

U drugom redu nalazi se  $N$  prirodnih brojeva manjih od 1 000 000 000 odvojenih razmakom. Ti brojevi označavaju redom koliko ovaca stane redom u prvi, drugi, ...,  $N$ -ti stroj za kloniranje.

U trećem redu nalazi se prirodni broj  $M$  ( $1 \leq M \leq N$ ), najveći broj strojeva koje Simona može navesti kod naredbe KLONIRAJ.



## **IZLAZNI PODACI**

Na izlaz je potrebno ispisati naredbe koje Simona treba izurлати, svaku u zaseban red.

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

## **PRIMJERI TEST PODATAKA**

|  |  |
|--|--|
| <pre>ulaz  3 2 3 6 2  izlaz  UNESI 2 KLONIRAJ 1 3 UNESI 3 KLONIRAJ 2 3</pre> | <pre>ulaz  5 25 25 30 25 25 3  izlaz  UNESI 2 KLONIRAJ 3 UNESI 5 KLONIRAJ 1 2 3 KLONIRAJ 5 4 1 KLONIRAJ 2 4 5 UNESI 3 KLONIRAJ 3</pre> |
|--|--|



Žitelji susjednih sela (nazovimo ih A i B) se nikako ne mogu dogovoriti o razgraničenju. Područje na kojem su smještena sela je pravokutno i možemo ga podijeliti na jedinične kvadrate zemlje raspoređene u R redaka i S stupaca. Na svakom kvadratu zemlje raste ili nekoliko stabala **jabuke** ili nekoliko stabala **kruške**.

Međunarodni posrednik pozvan je na medijaciju kako bi razriješio granični spor. Posrednik je odlučio da će kroz područje proći **bager** koji će **uništiti** sva stabla na kvadratima kroz koje prođe. Bager će krenuti iz kvadrata na krajnjem sjeverozapadu (prvi red, prvi stupac), kretati se kvadrat po kvadrat u smjeru **istoka ili jugoistoka ili juga** sve dok ne dođe u kvadrat na krajnjem jugoistoku (zadnji red, zadnji stupac). Selu A će pripasti svi kvadrati ispod uništenog područja, a selu B svi kvadrati iznad. Primijetite da je moguće da neko od sela ostane posve bez zemlje.

U svojoj mudrosti, posrednik je uočio da žitelji sela A više vole jabuke, a žitelji sela B više vole kruške. Stoga je odlučio da će bager prolaziti i rušiti tako da, nakon razgraničenja, **zbroj** broja stabala jabuka koje ostanu na području sela A i broja stabala krušaka koje ostanu na području sela B bude **najveći mogući**.

Napišite program koji određuje taj najveći mogući zbroj.

### ULAZNI PODACI

U prvom redu nalaze se prirodni brojevi R i S ( $2 \leq R, S \leq 1500$ ), dimenzije područja na kojem su smještena sela.

U svakom od idućih R redaka nalazi se po S opisa jediničnog kvadrata. Opis jediničnog kvadrata sastoji se od velikog slova 'J' (jabuke) ili 'K' (kruške) te broj stabala jabuka, odnosno krušaka na tom jediničnom kvadratu. Svako polje sadrži barem jedno, a najviše 99 stabala.

### IZLAZNI PODACI

U prvi i jedini red treba ispisati najveći mogući zbroj broja stabala jabuka koja će pripasti selu A i broja stabala krušaka koja će pripasti selu B.

### PRIMJERI TEST PODATAKA

|  |   |
|--|---|
| <b>ulaz</b><br><br>4 3<br>K2 K3 K5<br>J3 K1 J1<br>J2 J4 K1<br>K1 K3 J3<br><br><b>izlaz</b><br><br>17 | <b>ulaz</b><br><br>3 5<br>J5 J2 K3 J6 K2<br>J1 K20 J5 K3 K6<br>J3 J5 K3 K8 J3<br><br><b>izlaz</b><br><br>37 |
|--|---|

#### Pojašnjenje prvog primjera:

Bager se treba kretati redom: jugoistok, jugoistok, jug. Tada će ispod uništenog područja biti ukupno  $3+2+4=9$  stabala jabuke, a iznad  $3+5=8$  stabala kruške.



**DANI MLADIH INFORMATIČARA HRVATSKE 2009**  
**Dubrovnik, 22.-26. ožujka**

**SREDNJOŠKOLSKA SKUPINA, PASCAL/C/C++**  
**I. podskupina, Drugi dan natjecanja**  
**Zadatak PČELICE**

---

Pčelice, kao što svi znamo, imaju vrlo važnu ulogu kad se govori o oprašivanju cvijeća. Malo je, međutim, poznato da pčelice imaju vrlo čudan, jasno definiran protokol kojim odabiru cvijeće koje će oprašiti.

Na jednoj livadi nalazi se  $N$  cvjetova. Livadu predstavljamo kvadratom koji leži u koordinatnoj ravnini, a cvjetove točkama unutar nje. Nasuprotne točke koje određuju livadu su  $(0, 0)$  i  $(M, M)$ , a rub livade paralelan je s koordinatnim osima.

Roj pčela svaki dan odabire bazni cvijet i okuplja se oko njega. Četiri pčele ratnice zatim polijeću s baznog cvijeta svaka u jednom od četiri osnovna smjera: gore, dolje, lijevo i desno. Pčele ratnice zaustavljaju se kad nalete na neki drugi cvijet ili na rub livade.

Prostor koji će pčele taj dan oprašiti zadan je pravokutnikom čije su stranice paralelne s koordinatnim osima, a koji prolazi kroz sve četiri pčele ratnice koje ne puštaju nikog unutra, a ni van, dok posao nije obavljen. Dakle, taj dan će pčele oprašiti sve cvjetove unutar tog prostora; cvjetovi koji se nalazi na rubu pravokutnika neće biti oprašeni.

Napišite program koji će za zadane koordinate cvjetova **za svakog od njih** odrediti **ukupan broj oprašenih cvjetova** ako baš taj cvijet odaberemo za **bazni cvijet**.

### **ULAZNI PODACI**

U prvom redu nalazi se prirodan broj  $M$  ( $2 \leq M \leq 1\,000\,000$ ), dimenzije livade.

U drugom redu nalazi se prirodan broj  $N$  ( $1 \leq N \leq 300\,000$ ), broj cvjetova.

U sljedećih  $N$  redova nalaze se po dva prirodna broja  $x$  i  $y$  ( $0 < x, y < M$ ) odvojena razmakom, koordinate cvjetova.

Nijedan par cvjetova neće se nalaziti na istim koordinatama.



## IZLAZNI PODACI

U N redova potrebno je ispisati po jedan cijeli broj, ukupan broj oprušenih cvjetova ako određeni cvijet odaberemo za bazni, redom kojim su cvjetovi dani na ulazu.

## PRIMJERI TEST PODATAKA

|              |              |
|--------------|--------------|
| <b>ulaz</b>  | <b>ulaz</b>  |
| 3            | 10           |
| 4            | 7            |
| 1 2          | 1 3          |
| 2 2          | 5 3          |
| 2 1          | 7 3          |
| 1 1          | 4 2          |
| <b>izlaz</b> | 4 4          |
| 1            | 6 4          |
| 1            | 6 1          |
| 1            | <b>izlaz</b> |
| 1            | 3            |
|              | 5            |
|              | 3            |
|              | 5            |
|              | 3            |
|              | 3            |
|              | 5            |