

**Descrierea generală a problemelor**

<b>Nr. crt.</b>	<b>Denumirea problemei</b>	<b>Restricția referitoare la volumul utilizat de memorie</b>	<b>Restricția referitoare la timpul de execuție, secunde</b>	<b>Punctajul alocat problemei</b>
1.	Tetris	$\leq 256$ Mb	$\leq 2,0$	100
2.	Cursuri	$\leq 256$ Mb	$\leq 2,0$	100
3.	Orașe	$\leq 256$ Mb	$\leq 2,0$	100

*Notă.* În caz de egalitate de punctaj, prioritate se va da concurentului care a obținut punctajul respectiv primul.

## Tetris

Un joc video, Tetris, este reprezentat de o grilă rectangulară cu  $N$  rânduri și  $M$  coloane. Înainte ca jocul să înceapă, unele celule ale grilei sunt goale (indicate ca  $.$ ) și altele sunt umplute (indicate ca  $\#$ ). Pătratele umplute reprezintă un set de obiecte, iar pătratele umplute care sunt adiacente (orizontal sau vertical) ar trebui considerate a fi parte a aceluiași obiect rigid. În exemplul de mai jos avem 4 obiecte rigide (colorate în culori diferite):

```

. . # .
# # . #
. # # .
# . . .
# . . .

```

Când jocul începe, obiectele cad drept în jos pe grilă, toate cu aceeași viteză. Fiecare obiect continuă să cadă drept în jos până când atinge rândul de jos sau când o parte a sa aterizează direct deasupra altui obiect, moment în care se oprește.

**Sarcină.** Elaborați un program care determină starea finală a grilei.

**Date de intrare.** Prima linie a intrării standard conține două numere întregi  $N$  și  $M$  – numărul de rânduri și coloane a grilei. Următoarele  $N$  rânduri conțin a câte  $M$  caractere fiecare, descriind starea inițială a grilei. Dacă coloana  $j$  a rândului  $i$  al grilei conține un bloc, caracterul corespunzător din intrare va fi  $\#$ , în caz contrar va fi un caracter  $.$ .

**Date de ieșire.** Ieșirea standard va conține  $N$  linii, fiecare conținând  $M$  caractere, descriind starea finală a grilei. Dacă coloana  $j$  a rândului  $i$  al grilei conține un bloc, caracterul corespunzător în ieșire va fi  $\#$ , în caz contrar va fi un caracter  $.$ .

**Restricții.**  $1 \leq N, M \leq 10^6$ . Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în descrierea generală a problemelor propuse pentru rezolvare. Fișierul sursă va avea denumirea `tetris.pas`, `tetris.c` sau `tetris.cpp`.

### Exemplu 1.

*Intrare*

```

5 4
. . # .
# # . #
. # # .
# . . .
# . . .

```

*Ieșire*

```

. . . .
. . . .
# # # .
# # # .
# . . #

```

**Punctare:** Testele vor fi organizate astfel:

- Pentru 31% din teste,  $N \cdot M \leq 2000$ .
- Pentru alte 15% din teste,  $M = 2$ .
- Restul 54% din teste nu au restricții adiționale.

## Cursuri

Anul acesta Universitatea Tehnică a Moldovei a introdus cursuri extracurriculare pentru cei  $N$  studenți din anul I. Fiecare student trebuie să aleagă un curs din două disponibile. Convenționale vom nota că studentul  $i$  trebuie să aleagă unul din cele două cursuri disponibile  $x_i$  și  $y_i$ .

Pentru o mai bună organizare a cursurilor, conducerea universității a decis să grupeze studenții după  $M$  criterii. Un criteriu este descris prin 3 numere întregi:  $a$ ,  $b$  și  $c$ . Dacă  $a$  este egal cu 1, atunci studentul  $b$  nu poate frecventa același curs cu studentul  $c$ . Dacă  $a$  este egal cu 2, atunci studentul  $b$  va frecventa același curs cu studentul  $c$ .

**Sarcină.** Elaborați un program care fiind date cursurile pe care le poate frecventa fiecare student și criteriile de repartizare a studenților, stabilește o distribuție care corespunde celor  $M$  criterii.

**Date de intrare.** Prima linie a ieșirii standard conține două numere întregi separate prin spațiu,  $N$  și  $M$ . Următoarele  $N$  linii conțin două șiruri de caractere separate prin spațiu. Linia  $i$  conține șirurile  $x_i$  și  $y_i$ . Fiecare șir nu va avea mai mult de 20 de caractere. Următoarele  $M$  linii conțin trei numere întregi separate prin spațiu. Linia  $j$  conține numerele  $a_j$ ,  $b_j$ ,  $c_j$  ( $a_j = 1$  sau  $2$ ,  $1 \leq b_j, c_j \leq N$ ).

**Date de ieșire.** Ieșirea standard va conține  $N$  linii, unde linia  $i$  conține numele cursului ales de studentul  $i$ . Dacă nici o aranjare nu este posibilă, se va tipări un singur număr, -1.

**Restricții.**  $1 \leq N, M \leq 3 \cdot 10^5$ . Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în descrierea generală a problemelor propuse pentru rezolvare. Fișierul sursă va avea denumirea `cursuri.pas`, `cursuri.c` sau `cursuri.cpp`.

### Exemplu 1.

#### Intrare

```
4 4
informatica chimia
chimia informatica
informatica matematica
matematica informatica
2 1 2
1 1 4
1 3 4
1 1 3
```

#### Ieșire

```
chimia
chimia
informatica
matematica
```

**Explicație:** Observăm că doar studenții 1 și 2 pot frecventa împreună același curs. Ținând cont că disciplina informatica poate fi frecventată doar de studentul 3 sau 4, deducem că studenții 1 și 2 vor frecventa chimia. Studenții 3 și 4 nu pot frecventa același curs, respectiv una din posibilități este ca studentul 3 va frecventa informatica, iar studentul 4 matematica.

**Punctare:** Testele vor fi organizate astfel:

- Pentru 5% din teste,  $N \leq 3$  și  $M \leq 2$ ;
- Pentru 11% din teste,  $N \leq 10$  și  $M \leq 20$ ;
- Pentru 18% din teste,  $N \leq 20$  și  $M \leq 200$ ;
- Pentru 25% din teste,  $N \leq 1000$  și  $M \leq 4000$ ;
- Pentru restul 41% din teste, nu există restricții adiționale;

## Orașe

În Byteland, există  $N$  orașe numerotate de la 1 la  $N$ . În orașul  $i$  trăiesc  $A_i$  cetățeni. Între orașe există  $N - 1$  drumuri numerotate de la 2 la  $N$ . Drumul  $j$  leagă orașul  $j$  și orașul  $P_j$ , unde  $P_j < j$ . Există cel mult 36 de drumuri conectate la orice oraș.

În timpul iernii, toate drumurile vor fi transformate în autostrăzi cu sens unic din cauza condițiilor periculoase de conducere. Adică, drumul  $j$  va deveni o autostradă care fie este cu sens unic de la orașul  $j$  la orașul  $P_j$ , fie cu sens unic de la orașul  $P_j$  la orașul  $j$ . Fiecare cetățean dorește să trimită o carte de vacanță fiecărui alt cetățean. Fiind autostrăzile unicele căi de călătorie între orașe, cetățeanul  $x$  poate trimite o carte cetățeanului  $y$  doar dacă este posibil să călătorească din orașul în care locuiește cetățeanul  $x$  către orașul în care locuiește cetățeanul  $y$ , folosind doar autostrăzile.

**Sarcină.** Elaborați un program care determină numărul maxim de cărți de vacanță care pot fi trimise după transformarea tuturor drumurilor în autostrăzi.

**Date de intrare.** Prima linie a intrării standard conține un număr întreg  $N$ , numărul de orașe. A doua linie conține  $N$  numere întregi  $A_1, A_2, \dots, A_N$  separate prin spațiu – populația fiecărui oraș. A treia linie conține  $N-1$  întregi  $P_2, P_3, \dots, P_N$  separate prin spațiu – legătura orașelor 2, 3, ...,  $N$ .

**Date de ieșire.** Prima și singura linie a ieșirii standard va conține numărul maxim de cărți care pot fi trimise după transformarea tuturor drumurilor în autostrăzi.

**Restricții.**  $2 \leq N \leq 200\,000$ ;  $1 \leq A_i \leq 10\,000, i = 1, \dots, N$ ;  $1 \leq P_j \leq j, j = 2, \dots, N$ . Fie  $D$  numărul maxim de drumuri conectate la orice oraș. Se garantează că  $D \leq 36$ . Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în descrierea generală a problemelor propuse pentru rezolvare. Fișierul sursă va avea denumirea `orase.pas`, `orase.c` sau `orase.cpp`.

### Exemplu 1.

*Intrare*

```
4
3 3 4 1
1 2 1
```

*Ieșire*

```
67
```

**Explicație:** O modalitate posibilă de convertire a drumurilor în autostrăzi este ca drumul 2 să devină cu sens unic de la orașul 2 la orașul 1, drumul 3 să devină cu sens unic de la orașul 3 la orașul 2, iar drumul 4 să devină cu sens unic de la orașul 1 la orașul 4.

Fiecare cetățean din orașul 3 poate trimite 3 cărți de vacanță cetățenilor din orașul 3, 3 cărți de vacanță cetățenilor din orașul 2, 3 cărți de vacanță cetățenilor din orașul 1 și 1 carte de vacanță cetățeanului din orașul 4, pentru un total de 40 de cărți de vacanță trimise din orașul 3. Similar,

- cetățenii din orașul 2 trimit câte 6 cărți de vacanță fiecare, pentru un total de 18 cărți de vacanță.
- cetățenii din orașul 1 trimit câte 3 cărți de vacanță fiecare, pentru un total de 9 cărți de vacanță.
- cetățeanul din orașul 4 nu poate trimite nicio carte de vacanță.

În total, sunt trimise  $40 + 18 + 9 = 67$  de cărți de vacanță.

**Punctare:** Testele vor fi organizate astfel:

- Pentru 20% din teste,  $N \leq 10$ .
- Pentru alte 20% din teste,  $N \leq 1\,000$  și  $D \leq 10$ .
- Pentru alte 20% din teste,  $D \leq 18$ .
- Pentru alte 20% din teste, vor exista 37 de orașe, în care un oraș este conectat la alte 36 de orașe, iar aceste alte 36 de orașe sunt conectate doar la acest oraș.
- Restul 20% nu au restricții adiționale.