Problema 3 - operatii **100 puncte**

Fiind date două tablouri bidimensionale **a** şi **b**, cu **m** linii şi **n** coloane fiecare, definim următoarele operații:

1. suma tablourilor **a** şi **b**, ca fiind un tablou **c** cu **m** linii şi **n** coloane, în care fiecare element este egal cu suma elementelor de pe aceeşi linie şi coloană din **a** şi **b**. În acest caz folosim operatorul +, adică **c=a+b**.
2. produsul tablourilor **a** şi **b**, ca fiind un tablou **d** cu **m** linii şi **n** coloane, in care fiecare element este egal cu produsul elementelor de pe aceeşi linie şi coloană din **a** şi **b**. În acest caz folosim operatorul *, adică **d=a*b**. Dacă **a** şi **b** sunt tablouri identice (**a** şi **b** au elemente identice pe aceeaşi poziție), atunci pentru **d** se mai foloseşte şi notația $\mathbf{a}^2$ sau $\mathbf{b}^2$.

Exemplu:
Pentru **m**=2, **n**=3 şi tablourile:

a

1	5	7
2	15	3

b

2	10	100
1	8	0

Se obține

a+b

3	15	107
3	23	3

a*b

2	50	700
2	120	0

$\mathbf{a}^2$

1	25	49
4	225	9

$\mathbf{b}^2$

4	100	10000
1	64	0

Fiind dat un tablou bidimensional **a**, cu **m** linii, **n** coloane şi componente numere naturale dorim să determinăm un şir de tablouri bidimensionale: $\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_k$ cu număr minim de termeni (**k** minim), cu proprietatea că $\mathbf{a}=\mathbf{b}_1^2+\mathbf{b}_2^2+...+\mathbf{b}_k^2$.

Cerință
Să se scrie un program care determină tablourile bidimensionale $\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_k$ cu proprietatea din enunț.

Date de intrare
Fişierul de intrare `operatii.in` conține pe prima linie numerele naturale **m** şi **n** separate prin câte un spațiu. Pe umătoarele **m** linii se află elementele tabloului **a**, câte **n** numere pe o linie, în cadrul unei linii numerele fiind separate între ele prin câte un spațiu.

Date de ieşire
Fişierul de ieşire `operatii.out` conține pe prima linie un număr natural reprezentând valoarea **k**, apoi pe următoarele `k*m` linii elementele celor **k** tablouri $\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_k$. Fiecare dintre aceste tablouri va fi scris pe câte **m** linii consecutive, iar pe fiecare dintre aceste linii se vor afla câte **n** numere separate prin câte un spațiu.

Restricții şi precizări
- $1 \leq m, n \leq 200$
- Componentele tabloului **a** sunt numere naturale $\leq$ **10000**.
- Pot exista mai multe solutii, dar în fişierul de ieşire se va scrie una dintre ele.
- 30% din teste au componentele tabloului **a** mai mici sau cel mult egale cu **100** şi $m,n \leq 100$,
- 60% din teste au componentele tabloului **a** mai mici sau cel mult egale cu **1000.**

Exemplu

operatii.in	operatii.out	Explicație
2 3 1 2 4 5 5 9	2 1 1 0 2 2 3 0 1 2 1 1 0	**a** este: 1 2 4 5 5 9 $\mathbf{b}_1$ este: 1 1 0 iar $\mathbf{b}_1^2$ este: 1 1 0 2 2 3 4 4 9 $\mathbf{b}_2$ este: 0 1 2 iar $\mathbf{b}_2^2$ este: 0 1 4 1 1 0 1 1 0 Se observă că $\mathbf{b}_1^2 + \mathbf{b}_2^2 = \mathbf{a}$

Timp maxim de execuție/test: 1 secundă
Memorie totală disponibilă: 16 MB

Problema operatii **Baraj 2 juniori**
Autor: prof. Doru Popescu Anastasiu

Soluție

Pentru rezolvare mai întâi se determină pentru fiecare număr natural din mulțimea {0, 1, 2, ..., 10000} un mod de scriere ca sumă de cât mai puține numere pătrate perfecte.

Cu aceste numere folosite la descompuneri se formeaza un tablou. Acest tablou va avea 5 coloane, pentru că toate numerele de mai sus nu folosesc mai mult de 4 termeni în descompunere (o coloană pentru numărul de termeni din descompunere).

Pentru fiecare element din tabloul a, se determină câte numere se folosesc la descompunere. Numărul maxim dintre acestea este k.

Dacă un termen din a, a_{ij} are p numere folosite la descompunerea ca sumă de pătrate şi p<k, atunci în primele p tablouri se pun numerele din descompunere (radicalul lor), iar în următoarele valoarea 0.

Pentru exemplu avem

$0=0^2$
$1=1^2$
$2=1^2+1^2$
$3=1^2+1^2+1^2$
$4=2^2$
$5=1^2+2^2$
$6=1^2+1^2+2^2$
$7=1^2+1^2+1^2+2^2$
$8=2^2+2^2$
$9=3^2$
...

Dacă numerele folosite în tabloul a sunt 1, 2, 4, 5, 9. La 1 se foloseste un singur număr, la 2 două numere, la 4 un număr, la 5 două numere şi la 9 un număr, astfel se obține k=max(1,2), adică k=2.

După regula precizată se vor obtine tablourile:

b_1:
```
1 1 2
1 1 3
```
b_2:
```
0 1 0
2 2 0
```

În funcție de modul de determinarea a descompunerilor pentru toate numerele din mulțimea {0, 1, 2, ..., 10000}, algoritmul poate obține un punctaj mai mare sau mai mic decât 60 puncte.

1. Folosirea metodei backtracking, pentru fiecare număr din {0, 1, 2, ..., 10000}, conduce la un punctaj cel mult egal cu 60p.
2. Din observația că numerele de la 1 la 10000 se pot scrie ca sumă de cel mult patru pătrate, rezultă că putem determina toate descompunerile prin patru secvențe:
   a. Una cu un for până la 100 prin care determinăm numerele care se pot scrie ca sumă de un pătra perfect (care sunt chiar pătratele perfecte)
   b. Una prin care determinăm numerele care se pot scrie ca sumă de două pătrate perfecte (tot cu un for până la 100);
   c. Una prin care determinăm numerele care se pot scrie ca sumă de trei pătrate perfecte (cu două for-uri)
   d. Una prin care determinăm numerele care se pot scrie ca sumă de patru pătrate perfecte (cu trei for-uri).

   Astfel putem obține 100p.