

Problema 2018.2.3 – Mutarea firmei

Mobilierul și celelalte obiecte dintr-o firmă trebuie mutate cu mașinile într-o nouă locație. Toate obiectele au fost ambalate în cutii de diferite mărimi (volume), iar aranjarea lor în mașini se va face într-o manieră ce urmărește minimizarea numărului de mașini de transport necesare. Managerul de mutare știe că are un număr suficient de mașini astfel încât să încarce toate cutiile. În fiecare mașină încap cutii însumând în total un volum D . De asemenea, managerul de mutare știe câte cutii cu un anumit volum sunt de încărcat. Regulile de încărcare a mașinilor sunt:

- i. se încarcă pe rând câte o mașină;
- ii. se alege întotdeauna să se completeze mașina curentă cu cea mai mare cutie disponibilă;
- iii. se trece la următoarea mașină numai în condițiile în care în mașina curentă nu mai poate fi plasată nicio cutie dintre cele rămase și
- iv. toate cutiile au fost încărcate în mașini.

Cerință

Scrieți un program care îl poate ajuta pe managerul de mutare să aranjeze cutiile în mașini în mod eficient, conform regulilor enunțate mai sus.

Date de intrare

Se vor citi de la tastatură (fluxul *stdin*) următoarele date:

- de pe prima linie: două numere întregi D și k , separate prin spațiu, reprezentând D – volumul mașinilor exprimat în centimetri cubi, k – numărul de dimensiuni diferite pentru cutiile ce trebuie aranjate în mașini;
- de pe următoarele k linii: câte două numere întregi n și p , reprezentând numărul de cutii n de volum p centimetri cubi ce trebuie încărcate în mașini.

Cele k linii ce conțin informații despre cutii sunt date în ordinea inversă a volumului p .

Toate liniile conținând date de intrare sunt finalizate cu caracterul *newline* (tasta *Enter*).

Date de ieșire

Programul va afișa pe ecran (*stream*-ul standard de ieșire) m linii, corespunzătoare celor m mașini în care a fost încărcată cel puțin o cutie, în ordinea completării lor (conform regulilor). Fiecare dintre cele m linii va conține o serie de numere întregi, separate prin spațiu, reprezentând volumul cutiilor ce au fost plasate în acea mașină, în ordinea plasării lor în mașină (conform regulilor).

ATENȚIE la respectarea cerinței problemei: afișarea rezultatelor trebuie făcută EXACT în modul în care a fost indicat! Cu alte cuvinte, pe stream-ul standard de ieșire nu se va afișa nimic în plus față de cerința problemei; ca urmare a evaluării automate, orice caracter suplimentar afișat, sau o afișare diferită de cea indicată, duc la un rezultat eronat și prin urmare la obținerea calificativului „Respins”.

Restricții și precizări

1. Volumul mașinilor D este număr întreg în intervalul $[50; 10000]$.
2. Volumele cutiilor p sunt numere întregi în intervalul $[1; 1000]$.
3. Numerele n de cutii de diverse volume sunt numere întregi în intervalul $[1; 100]$.
4. Se garantează faptul că nu vor exista cutii de volum p mai mare decât volumul D al mașinilor.
5. Nu este necesar ca mașinile să fie umplute la volumul maxim D .

6. **Atenție:** În funcție de limbajul de programare ales, fișierul ce conține codul trebuie să aibă una din extensiile .c, .cpp, .java, sau .m. Editorul web **nu va adăuga automat** aceste extensii și lipsa lor duce la imposibilitatea de compilare a programului!
7. **Atenție:** Fișierul sursă trebuie numit de candidat sub forma: <nume>.<ext> unde nume este numele de familie al candidatului și extensia este cea aleasă conform punctului anterior. Atenție la restricțiile impuse de limbajul Java legate de numele clasei și numele fișierului!

Exemplu

Intrare	Ieșire
200 5	130 60
2 130	130 60
4 120	120 80
2 80	120 80
3 60	120 60
7 50	120 50
	50 50 50 50
	50 50

Explicație:
 Au fost completate 8 mașini astfel:
 Mașina #1:
 - cea mai mare cutie disponibilă (130 cm^3);
 - apoi cea mai mare cutie disponibilă (60 cm^3) care mai încapă în această mașină ($200 - 130 = 70 \text{ cm}^3$);
 - alte cutii nu mai încap în spațiul rămas: $70 - 60 = 10 \text{ cm}^3$.
 Mașina #2: identică cu mașina #1.
 Mașina #3:
 - cea mai mare cutie disponibilă (120 cm^3);
 - apoi cea mai mare cutie disponibilă (80 cm^3) care mai încapă în această mașină ($200 - 120 = 80 \text{ cm}^3$);
 - alte cutii nu mai încap în mașină – aceasta este completată în întregime.
 Mașina #4: identică cu mașina #3.
 Mașina #5:
 - cea mai mare cutie disponibilă (120 cm^3);
 - apoi cea mai mare cutie disponibilă (60 cm^3) care mai încapă în această mașină ($200 - 120 = 80 \text{ cm}^3$);
 - alte cutii nu mai încap în spațiul rămas: $80 - 60 = 20 \text{ cm}^3$.
 Mașina #6:
 - cea mai mare cutie disponibilă (120 cm^3);
 - apoi cea mai mare cutie disponibilă (50 cm^3) care mai încapă în această mașină ($200 - 120 = 80 \text{ cm}^3$);
 - alte cutii nu mai încap în spațiul rămas: $80 - 50 = 30 \text{ cm}^3$.
 Mașina #7:
 - cea mai mare cutie disponibilă (50 cm^3);
 - apoi, în ordine, cele mai mari cutii disponibile care mai încap în această mașină: 50, 50, 50;
 - alte cutii nu mai încap în mașină – acesta este umplută în întregime.
 Mașina #8:
 - cea mai mare cutie disponibilă (50 cm^3);
 - apoi, în ordine, cele mai mari cutii disponibile care mai încap în această mașină: 50;
 - au fost epuizate toate cutiile.

TimP de lucru: 120 de minute