

Problema 2018.2.4 – Încărcarea valizelor

Pentru plecarea în concediu familia a ales o mulțime de lucruri necesare ce trebuie împachetate în valize. Toate lucrurile au fost strânse și ambalate ca pachete de diferite mărimi și greutate, iar aranjarea lor în valize se va face într-un mod ce urmărește minimizarea numărului de valize necesare. Se știe că sunt suficiente valize astfel încât să încarce toate pachetele cu lucruri și haine. În fiecare valiză se pun pachete a căror greutate nu este mai mare decât D . Se știe de asemenea câte pachete cu o anumită greutate sunt de încărcat. Regulile de umplere a valizelor sunt:

- i. se umple pe rând câte o valiză;
- ii. se alege întotdeauna să se completeze valiza curentă cu cel mai mare pachet disponibil;
- iii. se trece la următoarea valiză numai în condițiile în care în valiza curentă nu mai poate fi plasat niciun pachet dintre cele rămase și
- iv. toate pachetele au fost puse în valize.

Cerință

Scrieți un program care poate ajuta familia să aranjeze pachetele în valize 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 – greutatea maximă de încărcare a unei valize, exprimată în kilograme, k – numărul de greutate diferite pentru pachetele ce trebuie aranjate în valize;
- de pe următoarele k linii: câte două numere întregi n și p , reprezentând numărul de pachete n de greutate p kilograme ce trebuie puse în valize.

Cele k linii ce conțin informații despre pachete sunt date în ordinea inversă a greutății 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 valize în care a fost pusă 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 greutatea cutiilor ce au fost puse în acea valiză, în ordinea plasării lor în valiză (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. Greutatea maximă admisă a valizelor D este număr întreg în intervalul $[10; 100]$.
2. Greutățile pachetelor p sunt numere întregi în intervalul $[1; 100]$.
3. Numerele n de pachete de diverse greutate sunt numere întregi în intervalul $[1; 100]$.
4. Se garantează faptul că nu vor exista pachete de greutate p mai mare decât greutatea maximă admisă D a valizelor.
5. Nu este necesar ca valizele să fie umplute la greutatea maximă 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
20 5 2 13 4 12 2 8 3 6 7 5	13 6 13 6 12 8 12 8 12 6 12 5 5 5 5 5 5 5
Explicație: Au fost completate 8 valize astfel: Valiza #1: <ul style="list-style-type: none"> - cel mai mare pachet disponibil (13 kg); - apoi cel mai mare pachet disponibil (6 kg) care mai încapă în această valiză ($20 - 13 = 7$ kg); - alte pachete nu mai încap în spațiul rămas: $7 - 6 = 1$ kg. Valiza #2: identică cu valiza #1. Valiza #3: <ul style="list-style-type: none"> - cel mai mare pachet disponibil (12 kg); - apoi cel mai mare pachet disponibil (8 kg) care mai încapă în această valiză ($20 - 12 = 8$ kg); - alte pachete nu mai încap în valiză – aceasta este completată în întregime. Valiza #4: identică cu valiza #3. Valiza #5: <ul style="list-style-type: none"> - cel mai mare pachet disponibil (12 kg); - apoi cel mai mare pachet disponibil (6 kg) care mai încapă în această valiză ($20 - 12 = 8$ kg); - alte pachete nu mai încap în spațiul rămas: $8 - 6 = 2$ kg. Valiza #6: <ul style="list-style-type: none"> - cel mai mare pachet disponibil (12 kg); - apoi cel mai mare pachet disponibil (5 kg) care mai încapă în această valiză ($20 - 12 = 8$ kg); - alte pachete nu mai încap în spațiul rămas: $8 - 5 = 3$ kg. Valiza #7: <ul style="list-style-type: none"> - cel mai mare pachet disponibil (5 kg); - apoi, în ordine, cele mai mari pachete disponibile care mai încap în această valiză: 5, 5, 5; - alte pachete nu mai încap în valiză – acesta este umplută în întregime. Valiza #8: <ul style="list-style-type: none"> - cel mai mare pachet disponibil (5 kg); - apoi, în ordine, cele mai mari pachete disponibile care mai încap în această valiză: 5; - au fost epuizate toate pachetele. 	

TimP de lucru: 120 de minute