

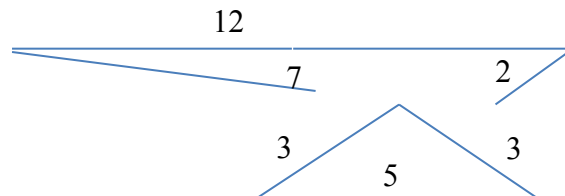
Problema 2 – triunghi

100 de puncte

Gigel este un pasionat al triunghiurilor. El colectează bețișoare de diferite lungimi și le assemblează în diferite triunghiuri. Ieri, el avea 6 bețișoare de lungimi 5, 2, 7, 3, 12 și 3. Din aceste bețișoare, Gigel a construit un triunghi de laturi 3, 3 și 5, iar bețișoarele de lungimi 2, 7, 12 au rămas nefolosite pentru că aceste lungimi nu pot forma laturile unui triunghi.

Din acest motiv, Gigel s-a hotărât să facă o colecție de bețișoare, dintre care oricum ar alege 3 elemente, acestea să nu poată forma laturile unui triunghi, proprietate pe care o vom numi în continuare *proprietate anti-triunghi*. Gigel, pornind de la setul inițial de lungimi 2, 7, 12, s-a gândit la două metode de realizare a unei colecții de 5 bețișoare cu proprietatea anti-triunghi, și anume:

1. Păstrează cel mai scurt bețișor, cel de lungime 2, și creează un set nou adăugând alte bețișoare de lungime mai mare sau egală cu cel inițial. De exemplu, următoarele 5 lungimi sunt corecte: 2, 2, 12, 50, 30.
2. Păstrează toate bețișoarele, și anume 2, 7, 12, pe care le va completa cu alte bețișoare de diferite lungimi (mai scurte sau mai lungi), astfel ca proprietatea anti-triunghi să se păstreze. Următoarele 5 lungimi respectă proprietatea anti-triunghi: 2, 7, 12, 4, 1.



Cerință

Cunoscând un șir de n numere naturale nenule $a_1, a_2, \dots, a_n$ având proprietatea anti-triunghi, și un număr k ($k > n$), se cere să construiești un șir de k numere naturale având proprietatea anti-triunghi, în conformitate cu una dintre următoarele două restricții:

Varianta 1. Cel mai mic element este identic cu cel mai mic element din șirul inițial.

Varianta 2. Printre cele k elemente ale șirului construit se regăsesc toate elementele șirului inițial.

Date de intrare

Fișierul de intrare `triunghi.in` conține pe prima linie valorile numerelor v , n și k , separate prin spațiu. Linia următoare conține n numere naturale separate prin spațiu, ce formează un șir cu proprietatea anti-triunghi.

Date de ieșire

Fișierul de ieșire `triunghi.out` va conține k numere pe o singură linie.

Dacă valoarea lui v este 1, atunci fișierul va conține k numere naturale cu proprietatea anti-triunghi, separate prin spațiu, în care cel mai mic element este identic cu minimul șirului dat în fișierul de intrare.

Dacă valoarea lui v este 2, atunci fișierul va conține k numere naturale cu proprietatea anti-triunghi, separate prin spațiu, printre care se regăsesc toate elementele șirului inițial.

Restricții și precizări

- $3 \leq n < k \leq 46$;
- $1 \leq \text{lungimea unui bețișor} \leq 2.000.000.000$;
- Pentru rezolvarea corectă a primei cerințe se acordă 30 de puncte, iar pentru cerința a doua se acordă 70 de puncte;
- Se garantează că întotdeauna există soluție;
- Soluția nu este unică - se admite orice răspuns corect.

Exemplu

triunghi.in	triunghi.out	Explicații
1 3 5 7 2 12	2 2 30 50 12	$v=1$, $n=3$, $k=5$. În varianta 1 avem de tipărit 5 numere, valoarea minimului este 2 în ambele șiruri.
2 3 5 7 2 12	1 4 12 7 2	$v=2$, $n=3$, $k=5$. În varianta 2 printre elementele șirului tipărit se regăsesc toate elementele șirului inițial.

Timp maxim de execuție: 0,1 secunde/test.

Memorie totală disponibilă 8 MB, din care 4 MB pentru stivă

Dimensiunea maximă a sursei: 5 KB.