

## 2 Formularea temei

Se citesc de la tastatura  $n$ ,  $m$  si  $3 \cdot n$  elemente care pot fi sau 0, sau cuprinse între 1 si  $n$ , unde se respecta conditia  $1 \leq n, m \leq 30$ . Se va genera **cea mai mica permutare in sens lexicografic** a multimii  $\{1, \dots, n\}$ , unde fiecare element apare **de exact 3 ori**, avand o distanta de **minimum**  $m$  elemente între oricare doua elemente egale, plecand de la anumite puncte fixe deja specificate.

De exemplu, pentru  $n = 5$ ,  $m = 1$  si secventa de 15 elemente

1 0 0 0 0 0 3 0 0 0 0 0 0 4 5

avem ca fiecare element din multimea  $\{1, 2, 3, 4, 5\}$  apare de 3 ori, si vrem sa fie cel putin  $m = 1$  element distanta între oricare doua elemente egale. Atunci, cea mai mica permutare in sens lexicografic, pastrand punctele fixe, este urmatoarea:

1 2 1 2 1 2 3 4 3 5 3 4 5 4 5

Se vor afisa la *standard output*, dupa caz,

- sau permutarea, daca exista, in formatul de mai sus: elementele se vor afisa cu spatii între ele pe ecran, iar la final recomandam sa afisati un caracter *backslash n*, in locul utilizarii *fflush*;
- sau  $-1$ , in cazul in care nu exista o permutare care sa satisfaca toate conditiile.

**IMPORTANT!** Pentru ca rezolvarile sa fie evaluate cu punctaj maxim, trebuie sa respecte urmatoarele cerinte:

1. sa fie utilizate procedurile, respectand **toate** conventiile prezentate in laborator, referitoare la constructia corecta a cadrului de apel;
2. algoritmul pe care il veti implementa va fi un algoritm **backtracking recursiv**.

In cazul in care implementati fara proceduri, veti primi doar 20% din punctajul pe care l-ati obtinut in urma evaluarii automate, iar in cazul in care implementati utilizand **backtracking iterativ** sau alti algoritmi, veti primi doar 70% din punctajul obtinut in urma evaluarii automate. Timpul maxim de rulare permis este de 2s.