

## 2 feladat – munte

100 pont

Adott egy  $x_1, x_2, \dots, x_n$ ,  $n$  természetes számokból álló sorozat. Egy maximális szomszédos elemekből álló  $x_i, x_{i+1}, \dots, x_{k-1}, x_k, x_{k+1}, \dots, x_j$  ( $1 \leq i < k < j \leq n$ ) alakú sorozatot amelyre igaz az  $x_i < x_{i+1} < \dots < x_{k-1} < x_k > x_{k+1} > \dots > x_j$  tulajdonság,  $x_k$  csúcsú hegynek nevezzük. Két hegy számsornak maximálisan egy közös eleme lehet az eredeti sorozatból. Egy hegy számsor maximálisan 3 elemből állhat. Például egy 3 4 6 8 értékekből álló sorozat nem tartalmaz egy hegy számsort sem, míg a 3 4 8 1 2 5 0 értékekből álló sorozat 2 hegy számsort tartalmaz: 3 4 8 1 és 1 2 5 0.

Az összes hegy számsor és azok csúcsainak a meghatározása után eltávolítjuk a sorozatból a hegy számsorok csúcsait és ezt az eljárást ismételten folytatjuk új hegy számsorok és azok csúcsainak meghatározásával az újjal kapott sorozatból. Az eljárás befejeződik ha a sorozatban már nem létezik egy hegy számsor sem.

### Feladat

Írjatok egy programot amely beolvassa az  $n, x_1, x_2, \dots, x_n$  számokat és azután meghatározza:

- az eredeti sorozatban található hegy számsorok számát;
- az összes hegy számsorok számát, amelyeket az eredeti sorozatból kiindulva kapunk, mindaddig, amíg a sorozatunk már nem tartalmaz egyetlen hegy számsort sem;
- a végső sorozatban levő elemek számát, amely már nem tartalmaz egyetlen hegy számsort sem.

### Bemenő adatok

A `munte.in` bemeneti állomány első sorában az  $n$  természetes szám található, a következő sorában pedig az  $x_1, x_2, \dots, x_n$  természetes számok egy-egy szóközzel elválasztva.

### Kimenő adatok

A `munte.out` kimeneti állomány első sorában az a) elvárásnak megfelelő természetes számot, a második sorában a b) elvárásnak megfelelő természetes számot, míg a harmadik sorában a c) elvárásnak megfelelő természetes számot tartalmazza.

### Megszorítások és pontosítások

- $3 \leq n \leq 100$
- $0 \leq x_i \leq 100000, 1 \leq i \leq n$
- Az a) elvárás helyes megoldásáért a pontszám 20%-ka érhető el.
- A b) elvárás helyes megoldásáért a pontszám 40%-ka érhető el.
- A c) elvárás helyes megoldásáért a pontszám 40%-ka érhető el.
- Az adott tesztekben biztosítjuk arról, hogy az eredeti számsorozat mindig tartalmaz legalább egy hegy számsort.

### Példa

| <code>munte.in</code> | <code>munte.out</code> | Magyarázat                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------------|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8<br>1 2 5 0 6 9 3 4  | 2<br>4<br>4            | a) Két hegy számsor van: 1 2 5 0 és 0 6 9 3<br>b) A csúcsok eltávolítása után az új sorozat 1 2 0 6 3 4 lesz. Ez a sorozat két hegy számsort tartalmaz: 1 2 0 és 0 6 3. A hegy számsorok csúcsainak eltávolítása után az új sorozat 1 0 3 4 lesz. Az új sorozat nem tartalmaz egyetlen hegy számsort sem. Összesen tehát 4 számsor van.<br>c) A végső sorozat 1 0 3 4, amelyben nincs már egy hegy számsor sem, 4 elemet tartalmaz. |

**Maximális futási idő:** 0.1 másodperc/teszt

**Rendelkezésre álló memória:** 2MB, amelyből 1MB a veremnek.

**A forráskód maximális mérete:** 5KB.