**Felvételi konzultáció: 2018 december 15**

**Minimum/maximum meghatározás; Rendezési módszerek**

**Gyakorló feladatok**

**1. Leghosszabb tömbszakasz, amely csak prímszámokat tartalmaz**

Írjatok algoritmust (alprogramot) amely egy természetes számokból álló számsorozatból meghatá­rozza azt a ***leghosszabb*** tömbszakaszt, amely ***csak prímszámokat*** tartalmaz. Bemeneti paraméterek: ***n*** – természetes szám ((10 ≤ ***n*** ≤ 10 000 000)) a számsorozat hossza, ***x*** – természetes számokat tároló tömb. A számok kisebbek, mint 30 000. Kimeneti paraméterek: ***start*** (a tömbszakasz első elemének indexe) és ***vége*** (a tömbszakasz utolsó elemének indexe).

***Példa:*** ha ***n*** = 10 és ***x*** = (6, 3, 5, 8, 1, 2, 3, 7, 19, 12), akkor ***start*** = 6 és ***vége*** = 9.

**2. Szigetek (felvételi, 2016)**

Egy légitársaság az utasok rendelkezésére bocsátotta azoknak a földrajzi pontoknak a magasságait tar­talmazó sorozatot, amelyek fölött a repülőgép száll majd Kolozsvár és New York között. Az ***a*** sorozat­nak ***n*** darab (3 ≤ ***n*** ≤ 10 000), 30 000-nél szigorúan kisebb természetes szám eleme van. A szárazföld­nek megfelelő pontok magasságai 0-tól különböznek, míg az óceánnak megfelelő pontok magasságai egyenlők 0-val. *Sziget*nek olyan egymás utáni szárazföldnek megfelelő pontok sorozatát nevezzük, amely előtt és után víz található. Az alábbi követelmények teljesítésére írjatok egy-egy algoritmust (alprogramot):

1. Határozzátok meg a szigetek számát.
2. Határozzátok meg a leghosszabb sziget kezdetét, valamint a végét jelző pont sorszámát. Ha több megoldás létezik, csak egyet kell meghatároznotok. Ha nem létezik egyetlen sziget sem, akkor az eredmény 0 0.
3. Döntsétek el, hogy a leghosszabb sziget *hegy* típusú-e vagy sem. Egy sziget *hegy* típusú, ha a felszínén található magasságok egy adott elemig szigorúan növekvő – nem üres – sorozatot alkotnak, majd szigorúan csökkenőt, amely nem üres.
4. Döntsétek el, hogy a szárazföldnek megfelelő pontok magasságai páronként *különböző értékűek* vagy sem.
5. Ha a 4. pontban feltett kérdésre a válasz „*nem*”, határozzátok meg a leggyakoribb magasság értékét és az előfordulásainak számát. Ha több ilyen magasság létezik, csak egyet kell meg­adnotok.

**1. *Példa:*** ha ***n*** = 15 és ***a*** = (10, 2, 1, 0, 7, 0, 1, 2, 13, 5, 0, 0, 8, 5, 2), összesen 2 sziget van, a leghosszabb sziget a 7. és 10. sorszámú magasságok között található. A legnagyobb sziget *hegy* típusú. A magas­ságok értékei nem különbözők, és a leggyakoribb magasság értéke 2, amely 3-szor fordul elő.

**2. *Példa:*** ha ***n*** = 10 és ***a*** = (1, 2, 0, 1, 2, 13, 0, 0, 1, 2), egyetlen sziget létezik, amely a 4. és 6. sorszámú pontok között található és *nem* *hegy* típusú. A magasságok értékei nem különbözők, az egyik leg­gyakoribb magasság értéke 1 és 3-szor fordul elő.

**3. Nyeregpontok**

Adott egy ***n*** soros és ***m*** oszlopos (0 < ***n***, ***m*** ≤ 100), egész számokat tároló kétdimenziós ***a*** tömb. Írjatok algoritmust, amely megszámolja a tömb nyeregpontjait! Egy ***aij*** elemet *nyeregpont*nak nevezünk, ha az ***aij*** elem legnagyobb a ***j***. oszlopban és legkisebb az ***i***. sorban, és fordítva. Az algoritmus bemeneti paraméterei ***n*** és ***a***, kimeneti paraméterek a nyeregpontok darabszáma (***k***) és az indexeik.

***Példa:*** Ha ***n*** = 2, ***m*** = 6 és ***a*** = akkor ***k*** = 2, indexek: (1, 2) és (2, 5).

**4. Maximális szorzat (felvételi, 2016)**

Adott az ***n*** elemű (3 ≤ ***n*** ≤ 10 000), egész számokat tároló ***x*** sorozat, ahol az elemek értéke nagyobb, mint -30 000 és kisebb, mint 30 000. Írjatok alprogramot, amely meghatároz *három* számot az ***x*** sorozatból, amelyeknek a szorzata *maximális*. Az alprogram bemeneti paraméterei ***n*** és ***x***, kimeneti paraméterei az ***a***, ***b*** és ***c*** számok, amelyek az ***x*** sorozat elemei és rendelkeznek a kért tulajdonsággal. Ha a feladatnak több megoldása van, csak egyet kell megadnotok.

***Példa:*** ha ***n*** = 10 és ***a*** = (3, -5, 0, 5, 2, -1, 0, 1, 6, 8), a három szám: ***a*** = 5, ***b*** = 6, ***c*** = 8.