

II. TÉTEL

(40 pont)

1. Adott a mellékelt algoritmus pszeudokódban.

- Írja le a kiírt értéket, ha a beolvasott számok, ebben a sorrendben 8, 3, 1, 5, 2, 4, 5, 5, 7. (6p.)
- Ha az n változóba beolvasott érték 3, írjon egy adatsort amit a továbbiakban beolvashatunk úgy, hogy az algoritmus végrehajtása során a kiírt érték 0 legyen. (6p.)
- Írja meg az adott algoritmusnak megfelelő C/C++ programot. (10p.)
- Írjon az adott algoritmussal egyenértékű pszeudokód algoritmust, amelyben az ismétlődő szerkezetet egy **minden...végezd el** szerkezettel helyettesíti. (6p.)

```

beolvas n
(nemnulla természetes szám)
s ← 0; x ← 0; p ← 1
ismételd
    beolvas y (nemnulla természetes szám)
    ha y > x akkor
        s ← s + y; x ← y
    különben
        ha x = y akkor s ← s - y
    p ← p + 1
ameddig p > n
    kiír s
    
```

- Annak ellenőrzésére, hogy az (5, 7, 15, 19, 21, 23, 25) egydimenziós tömbben létezik-e az x értékű elem, a bináris keresés módszerét használjuk. Írja le az x összes lehetséges egész értékének halmazát úgy, hogy a tömb elemeinek sorozata, amelyekkel össze lesz hasonlítva az x értéke a megadott módszer alkalmazása során 19, 23, 21 legyen. (6p.)
- A p és r egész típusú változók egy számtani haladvány első tagját, valamint a léptékét tárolják. Tudva azt, hogy a haladvány egy tagja és az előző tag különbsége egyenlő a léptékkel, írjon egy C/C++ utasítást, amely a t egész típusú változónak értékül adja a haladvány 3-ik tagját. (6p.)

III. TÉTEL

(30 pont)

- Egy természetes értékhez rendelt **egységes számok** nevezzük azt a belőle kapott számot, amelyet úgy állítunk elő, hogy vagy eltávolítjuk az összes páros számjegyet vagy az összes páratlan számjegyet.
Példa: a 19472 számhoz a 197 és 42 egységes számokat rendeljük hozzá.
Beolvasunk egy n természetes számot, amelynek minden számjegye nullától különböző, és amelynek van legalább egy páros és legalább egy páratlan számjegye. Ki kell írni egy olyan számot, amelyet úgy kapunk, hogy az n -hez rendelt páratlan egységes szám számjegyeit az n -hez rendelt páros egységes szám számjegyei követik, mint a példában.
Írjon pszeudokód algoritmust az adott feladat megoldására.
Példa: ha $n=19472$, akkor a kiírt érték 19742. (10p.)
- Írjon egy C/C++ programot, amely a billentyűzetről beolvas egy n ($n \in [2, 20]$) természetes számot, majd egy egydimenziós tömb n elemét, amelyek nullától különböző egész számok a $[-10^9, 10^9]$ intervallumból. A program a képernyőre az **alterneaza** üzenetet írja ki, ha a tömbben nincs két pozitív szám egymást követő pozíciókon, vagy két negatív szám egymást követő pozíciókon, vagy a **nu alterneaza** üzenetet ellenkező esetben.
Példa: ha $n=5$ és a tömb (-32, 15, -7, 16, -20), a kiírt üzenet **alterneaza**
valamint, ha $n=5$ és a tömb (-32, 15, -7, 16, 20), a kiírt üzenet **nu alterneaza** (10p.)
- Az idő során egy személy több sakk és go versenyen vett részt. **Motiváló eredmény** volt az, amely esetén jobb helyezést ért el (kisebb szám) a versenyen, mint az azonos típusú előző versenyen.
A **bac.txt** állomány leg több 10^4 természetes számot tartalmaz, amelyek a versenyeken elért eredmények időrendi sorrendben: minden versenyeredmény esetén egy számpár, amelyben a versenyen elért helyezést (egy szám az $[1, 300]$ intervallumból), az 1 érték követ sakkverseny esetén, vagy a 2 érték go verseny esetén. Minden számpár az állomány egy külön sorában található, és az értékeket egy szököz választja el.
Ki kell írni a képernyőre azon **sakk** versenyeken elért eredmények számát, amelyek motiváló eredmények voltak. Tervezzon a futási idő és a felhasznált memória szempontjából hatékony algoritmust.
Példa: ha az állomány a mellékelt számokat tartalmazza, a képernyőre kiírt érték 3 mert sakkversenyen három motiváló eredményt ért el (3. helyezés a 7. helyezés után, 1. helyezés a 5. helyezés után és 1. helyezés a 2. helyezés után).
a. Írja le saját szavaival a használt algoritmust, és indokolja annak hatékonyságát. (2p.)
b. Írja meg a leírt algoritmusnak megfelelő C/C++ programot. (8p.)

```

6 1
8 2
7 1
5 2
3 1
4 1
5 1
1 2
4 2
1 1
2 1
1 1
    
```