

SUBIECTUL al II-lea

(40 de puncte)

1. Se consideră algoritmul alăturat, reprezentat în pseudocod.

- Scriveți valoarea afișată dacă se citesc, în această ordine, numerele 8, 3, 1, 5, 2, 4, 5, 5, 7. (6p.)
- Dacă pentru n se citește numărul 3, scrieți un set de date care pot fi citite în continuare, astfel încât, în urma executării algoritmului, să se afișeze 0. (6p.)
- Scriveți programul C/C++ corespunzător algoritmului dat. (10p.)
- Scriveți în pseudocod un algoritm, echivalent cu cel dat, înlocuind structura repetitivă cu o structură de tip **pentru...execută**. (6p.)

```

citește n
    (număr natural nenul)
s ← 0; x ← 0; p ← 1
repetă
    citește y (număr natural nenul)
    dacă y > x atunci
        s ← s + y; x ← y
    altfel
        dacă x = y atunci s ← s - y
    p ← p + 1
până când p > n
    scrie s
    
```

- Pentru a verifica dacă în tabloul unidimensional (5, 7, 15, 19, 21, 23, 25) există elementul cu valoarea x , se aplică metoda căutării binare. Scrieți mulțimea tuturor valorilor întregi posibile ale lui x , astfel încât succesiunea de elemente ale tabloului a căror valoare se compară cu valoarea lui x pe parcursul aplicării metodei indicate să fie 19, 23, 21. (6p.)
- Variabilele întregi p și r memorează primul termen, respectiv rația unei progresii aritmetice. Știind că diferența dintre un termen al progresiei și termenul anterior este egală cu rația, scrieți o instrucțiune C/C++ prin care i se atribuie variabilei întregi t valoarea celui de al 3-lea termen al progresiei. (6p.)

SUBIECTUL al III-lea

(30 de puncte)

- Numim **număr uniform** asociat unei valori naturale numărul obținut din aceasta prin eliminarea fie a tuturor cifrelor sale pare, fie a tuturor cifrelor sale impare.

Exemplu: lui 19472 i se asociază numerele uniforme 197 și 42.

Se citește un număr natural cu toate cifrele nenule, n , având cel puțin o cifră pară și cel puțin o cifră impară, și se cere să se scrie un număr obținut din cifrele numărului uniform impar asociat lui n , urmate de cifrele numărului uniform par asociat lui n , ca în exemplu.

Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.

Exemplu: dacă $n=19472$, se scrie numărul 19742.

(10p.)

- Scriveți un program C/C++ care citește de la tastatură un număr natural, n ($n \in [2, 20]$), apoi cele n elemente ale unui tablou unidimensional, numere întregi nenule din intervalul $[-10^9, 10^9]$. Programul afișează pe ecran mesajul **alterneaza** dacă în tablou nu există două numere pozitive sau două numere negative pe poziții consecutive, sau mesajul **nu alterneaza** în caz contrar.

Exemplu: pentru $n=5$ și tabloul (-32, 15, -7, 16, -20) se afișează mesajul **alterneaza**

iar pentru $n=5$ și tabloul (-32, 15, -7, 16, 20) se afișează mesajul **nu alterneaza**

(10p.)

- De-a lungul timpului, o persoană a participat la mai multe concursuri de șah și de go. **Un rezultat motivant** corespunde unei poziții mai bune (cu un număr mai mic) în clasamentul unui concurs, comparativ cu poziția obținută la concursul anterior de același tip.

Fișierul **bac.txt** conține un șir de cel mult 10^4 numere naturale, reprezentând rezultatele obținute la concursurile la care a participat, în ordine cronologică: pentru fiecare dintre rezultate, câte o pereche formată din poziția ocupată în clasamentul concursului corespunzător (un număr din intervalul $[1, 300]$), urmată de valoarea 1, dacă acesta a fost concurs de șah, sau de valoarea 2, dacă acesta a fost concurs de go. Fiecare pereche se află pe câte o linie a fișierului, cu valorile separate printr-un spațiu.

Se cere să se afișeze pe ecran numărul de rezultate motivante obținute la concursurile de **șah**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.

Exemplu: dacă fișierul conține numerele alăturate, pe ecran se afișează 3

pentru că la șah a obținut trei rezultate motivante (poziția 3 după poziția 7, poziția 1 după poziția 5 și poziția 1 după poziția 2).

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia.

(2p.)

b. Scrieți programul C/C++ corespunzător algoritmului proiectat.

(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
    
```