

SUBIECTUL al II-lea

(40 de puncte)

1. Se consideră algoritmul alăturat, reprezentat în pseudocod.

- Scieți 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, scieți un set de date care pot fi citite în continuare, astfel încât, în urma executării algoritmului, să se afișeze 0. (6p.)
- Scieți programul C/C++ corespunzător algoritmului dat. (10p.)
- Scieți 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
    
```

- Utilizând metoda backtracking se generează toate seturile de câte cel mult trei materiale de promovare turistică din mulțimea ordonată {broșură, ghid, pliant, revistă}, astfel încât din același set să nu facă parte simultan un pliant și o revistă. Două seturi sunt distincte dacă diferă prin cel puțin un material. Primele cinci soluții generate sunt, în această ordine, (broșură), (broșură, ghid), (broșură, ghid, pliant), (broșură, ghid, revistă), (broșură, pliant). Scieți a șasea și a șaptea soluție generată. (6p.)

- Variabilele i și j sunt de tip întreg, iar variabila a memorează un tablou bidimensional cu 5 linii și 5 coloane, numerotate de la 0 la 4, având inițial toate elementele nule.

Fără a utiliza alte variabile, scieți secvența C/C++ de mai jos, înlocuind punctele de suspensie cu instrucțiuni adecvate, dintre care cel mult patru atribuirii, astfel încât, în urma executării secvenței obținute, tabloul memorat în variabila a să aibă elementele alăturate.

```

for (i=0; i<5; i++)
    for (j=0; j<5; j++)
    
```

.....

(6p.)

0	9	10	19	20
1	8	11	18	21
2	7	12	17	22
3	6	13	16	23
4	5	14	15	24

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.

Subprogramul **ImparPar** are un singur parametru, n , prin care primește un număr natural cu toate cifrele nenule ($n \in [11, 10^9]$), având cel puțin o cifră pară și cel puțin o cifră impară. Subprogramul returnează un număr obținut din cifrele numărului uniform impar asociat lui n , urmate de cifrele numărului uniform par asociat lui n , ca în exemplu. Scieți definiția completă a subprogramului.

Exemplu: dacă $n=19472$, subprogramul returnează numărul 19742.

(10p.)

- Scieți un program C/C++ care citește de la tastatură un cuvânt format din cel puțin două și cel mult 100 de caractere, numai litere mici ale alfabetului englez, și afișează pe ecran mesajul **alterneaza**, dacă în text nu există două vocale sau două consoane pe poziții consecutive, sau mesajul **nu alterneaza**, în caz contrar.

Exemplu: pentru cuvintele **imunopolizaharide** sau **pas** se afișează mesajul **alterneaza**

iar pentru cuvântul **aer** 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, separate printr-un spațiu, două valori naturale, reprezentând, în această ordine, numărul de rezultate motivante obținute la concursurile de șah, respectiv la cele de go. 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 2

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), iar la go două rezultate motivante (poziția 5 după poziția 8, poziția 1 după poziția 5).

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

(2p.)

b. Scieți 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