

(30 de puncte)

Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul întreg memorat în variabila întreagă **x** aparține mulțimii {−2, −1, 1, 2}. (4p.)
- a. **abs(x)>2 || x==0** b. **abs(x)<=2 && x!=0**
- c. **abs(x-2)<1** d. **abs(x-1)>2**
2. Variabilele **x**, **y** și **d** sunt de tip întreg și memorează câte un număr natural strict pozitiv. Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila **d** să memoreze cel mai mare divisor comun al valorilor memorate în variabilele **x** și **y**. (4p.)
- a. **x%d+y%d!=0** b. **x%d!=y%d**
- c. **(x+y)%d!=0** d. **(x*d)*(y*d)!=0**
- ```
d=x;
if(d>y) d=y;
while(.....)
 d=d-1;
```

**Scrieti pe foaia de examen raspunsul pentru fiecare dintre cerintele urmatoare.**

3. Se consideră două puncte **A** și **B** din planul definit de sistemul de coordonate **xOy**. Coordonatele punctului **A** sunt memorate în variabilele întregi **xa** (abscisa) și **ya** (ordonata), iar coordonatele punctului **B** sunt memorate în variabilele întregi **xb** (abscisa) și **yb** (ordonata). Niciunul dintre cele două puncte nu se află în originea sistemului. Scrieți o expresie C/C++ care să aibă valoarea **1** dacă și numai dacă punctul **A** este situat pe axa **Oy**, iar punctul **B** este situat pe axa **Ox** a sistemului de coordonate. (6p.)
4. Se citește un număr natural, **n** ( $n \geq 1$ ), și se cere să se scrie numărul obținut prin duplicarea fiecărei cifre pare a lui **n** sau **-1** dacă acesta nu are nicio cifră pară. **Exemplu:** dacă **n=2380** se scrie **2238800**. (10p.)
- a) Scrieți, în pseudocod, algoritmul de rezolvare pentru problema enunțată. (10p.)
- b) Precizați rolul tuturor variabilelor care au intervenit în algoritmul realizat la punctul a) și indicați datele de intrare, respectiv datele de ieșire ale problemei enunțate. (6p.)

**SUBIECTUL al III-lea**

**(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Tablourile unidimensionale **A** și **B** au elementele: **A**=(20,17,12,10,3), iar **B**=(45,16,12,7,2). În urma interclasării lor în ordine descrescătoare se obține tabloul cu elementele: **(4p.)**
- a. (45,17,12,10,3)                      b. (45,20,16,17,12,12,7,10,2,3)
- c. (45,20,17,16,12,12,10,7,3,2)      d. (45,20,17,16,12,12,7,10,3,2)

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. În secvența alăturată toate variabilele sunt de tip întreg. Scrieți secvența, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei **s** să fie egală cu suma numerelor **întregi** citite care au cel mult două cifre. **(6p.)**
- ```
s=.....;
for(i=1;i<=10;i++)
{ cin>>x; | scanf("%d",&x);
  if(.....)
    s=s+x;
}
```
3. Scrieți un program C/C++ care citește de la tastatură un număr natural, **n** ($n \in [2, 10^2]$), și numere naturale din intervalul **[0, 1000]**, în această ordine: cele **n** elemente ale unui tablou unidimensional și un număr **x**. Prima poziție a tabloului este 0. Programul afișează pe ecran mesajul **DA**, dacă există cel puțin un element egal cu **x** aflat pe pozițiile impare ale tabloului sau mesajul **NU** în caz contrar. **Exemplu:** pentru **n=5**, tabloul (1, 3, 4, 2, 6) și **x=2**, se afișează pe ecran mesajul **DA** iar pentru **n=5**, tabloul (1, 12, 2, 4, 6) sau tabloul (1, 3, 12, 4, 6) și **x=2**, se afișează pe ecran mesajul **NU** **(10p.)**
4. Fișierul **bac.txt** conține un șir de cel mult 10^6 numere naturale distincte din intervalul **[0, 10^9]**. Numerele din șir sunt separate prin câte un spațiu. Se cere să se determine ultimii doi termeni pari din șirul aflat în fișier care sunt precedați de doar un termen impar. Termenii determinați se afișează pe ecran, în ordinea apariției lor în șir, separați printr-un spațiu, iar dacă în șir nu există doi astfel de termeni, pe ecran se afișează mesajul **Nu exista**. Pentru determinarea termenilor ceruți se utilizează un algoritm eficient din punctul de vedere al spațiului de memorie și al timpului de executare. **Exemplu:** dacă fișierul conține numerele **4 16 5 12 14 30 212 16 11 400** se afișează pe ecran **212 16** (**4** și **16** sunt precedate de 0 numere impare, **12**, **14**, **30**, **212** și **16** sunt precedate de 1 număr impar, iar **400** este precedat de 2 numere impare; dintre numerele **12**, **14**, **30**, **212** și **16** ultimele două sunt **212** și **16**).
a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**
b) Scrieți programul C/C++ corespunzător algoritmului descris. **(6p.)**