

Proba E. d)
INFORMATICĂ
Limbajul C/C++

Varianta 1

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică *Filieră vocațională, profil militar, specializare matematică-informatică*

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- Identificatorii utilizați în rezolvări trebuie să respecte precizările din enunț (bold), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.
- În grafurile din cerințe oricare muchie are extremități distincte și oricare două muchii diferă prin cel puțin una dintre extremități.

SUBIECTUL I

(20 de puncte)

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

1. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă valoarea variabilei întregi **m** este un multiplu comun al variabilelor întregi strict pozitive **a** și **b**.

a. **`a%m==0 && b%m==0 && a*b`**
c. **`!(a%m+b%m) && a*b`**

b. **`!(m%a) && !(m%b) && a*b`**
d. **`m%a==0 || m%b==0 && a*b`**

2. Subprogramul **f** este definit alăturat. Indicați valoarea pe care o poate avea parametrul **n**, astfel încât, în urma apelului, **f(n)** să aibă valoarea 5.

```
int f(long n)
{ if(n==0) return 0;
  else
    if(n%10%2!=0) return 1+f(n/10);
    else return f(n/10);
}
```

a. **12345** b. **117755** c. **88665501** d. **133258780**

3. În secvența alăturată, variabila **A** memorează elementele întregi ale unei matrice cu **n=5** linii și **n=5** coloane (numerotate începând de la 1), restul variabilelor fiind de tip întreg. Care va fi suma elementelor situate pe ultima linie a matricei la finalul executării secvenței?

```
for(i=1;i<=n;i++)
  for(j=1;j<=n;j++)
    if(i<j) A[i][j]=i;
    else A[i][j]=j;
```

a. **11** b. **15** c. **25** d. **14**

4. Se consideră următoarele declarații:

```
struct elev
{ char nume[21];
  int nr;
  float medii[14];
};
```

elev e[25], x;

Care dintre următoarele instrucțiuni este corectă?

a. **`e.medii=x.medii;`** b. **`e[2].nume=x;`** c. **`x.medii[3]=e[2].medii[3];`** d. **`x.medii=7.5;`**

5. Un graf neorientat are 6 noduri, numerotate de la 1 la 6, și muchiile [1,3], [2,3], [2,5], [3,5], [4,6]. Numărul minim de muchii ce trebuie adăugate pentru ca graful să devină conex este:

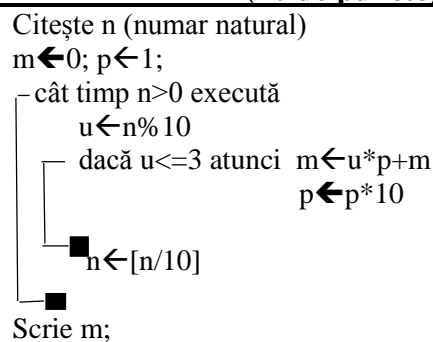
a. **1;** b. **2;** c. **3;** d. **0.**

SUBIECTUL II

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

- a. Scrieți ce se afișează în urma executării algoritmului dacă se citește pentru n valoarea 8172039? (6p.)
- b. Dați exemplu de o valoare care trebuie citită pentru n astfel încât algoritmul să afișeze o valoare egală cu cea citită. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat (10p.)
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind structura cât timp cu o structură repetitivă cu test final. (6p.)



2. Utilizând metoda backtracking se generează toate șirurile ordonate alfabetic care conțin cel puțin trei litere distincte, din mulțimea {F, O, R, M, A}, astfel încât în orice șir generat nu există două vocale alăturate. Primele patru soluții generate sunt, în această ordine: AFM, AFMO, AFMOR, AFMR. Scrieți următoarele trei soluții generate imediat după soluția AMOR. (6p.)

3. În secvența de mai jos, A este o matrice cu elemente întregi, cu 10 linii și 10 coloane în care liniile și coloanele sunt indexate de la 0. Completați secvența dată astfel încât în urma executării ei, să se afișeze numărul de elemente strict negative de pe linia X, elemente aflate pe coloane cu numere de ordine divizibile cu 3. (6p.)

```

k ← .....
for (i=0; i<=.....; i++)
    if (.....) k++;
cout<<k; | printf("%d ", k);
    
```

SUBIECTUL al III-lea

(30 de puncte)

1. Se consideră subprogramul **dublu**, cu parametrul **n**, prin care primește o valoare naturală nenulă $n < 10^9$ și returnează numărul obținut prin dublarea cifrei din mijloc dacă are număr impar de cifre sau -1 în caz contrar. Scrieți definiția completă a subprogramului.

Exemplu: dacă $n=70613$ în urma executării subprogramului se va returna **706613** (10p.)

2. Scrieți un program C/C++ care citește de la tastatură un text cu cel mult **255** caractere format din cuvinte (litere mici și litere mari) și numere reale separate printr-unul sau mai multe spații. Câte numere conține textul și câte cifre s-au citit. Numerele obținute se vor afișa cu spațiu între ele.

Exemplu: Dacă textul citit este "Valoarea absoluta a lui -12 este 12 iar valoarea lui PI este 3,14" Răspunsul afișat este 3 7 (10p.)

3. Fisierul **numere.in** memorează numere naturale astfel:

- pe prima linie o valoare naturală **n** ($n \leq 100$)
- pe a doua linie **n** valori naturale, de cel mult 9 cifre, ordonate crescător, separate prin spațiu
- pe a treia linie o valoare naturală **m** ($m \leq 1000000$)
- pe a patra linie **m** valori naturale, de cel mult 9 cifre, ordonate crescător, separate prin spațiu

Se cere să se afișeze crescător toate valorile distincte, din cele două șiruri, divizibile cu 3. Dacă nu există valori divizibile cu 3 se va afișa mesajul **nu exista**. Pentru determinarea valorilor cerute se utilizează un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fisierul **numere.in** conține:

```

10                                     Pe ecran se va afișa:
8 13 19 24 36 52 63 84 108 135       6 15 24 36 63 84 108 135
10
6 15 20 24 50 56 63 68 70 71
    
```

a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. (2p.)

b) Scrieți programul C/C++ corespunzător algoritmului descris. (8p.)