



EXAMENUL DE BACALAUREAT - 2024

Simulare județeană

Proba E. d) Informatică C/C++

Filiera teoretică – Profilul real; Specializarea matematică-informatică și matematică-informatică intensiv informatică

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei ore.
- În rezolvările cerute, identificatorii utilizați 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).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară

Subiectul I (20p)

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. Care dintre următoarele expresii este adevărată dacă și numai dacă, variabilele x și y conțin două numere naturale impare, consecutive :
 - a. $x \% 2 == 1 \ \&\& \ y \% 2 == 1$
 - b. $x \% 2 != 0 \ \&\& \ x - y == 2$
 - c. $x \% 2 == 1 \ \&\& \ x - y == 2 \ || \ y - x == 2$
 - d. $x \% 2 != 0 \ \&\& \ !(x - y != 2 \ \&\& \ y - x != 2)$
2. Se consideră un graf neorientat complet, cu 40 de vârfuri. Care este numărul maxim de muchii care pot fi eliminate, astfel încât graful să fie eulerian și hamiltonian?
 - a. 40
 - b. 760
 - c. 740
 - d. 780
3. Se dă un arbore cu rădăcina, cu 8 noduri, prin vectorul de tați următor: (3, 4, 4, 0, 2, 3, 6, 6).

Câți descendenți are nodul 3 ?

 - a. 4
 - b. 2
 - c. 3
 - d. 1

4. Fie următoarea funcția recursivă:

```
void f(int k)
{
    if(k>0){
        cout<<k;
        if (k%3==0) {cout<<'*'; f(k/3);}
        else {f(k/2); cout<<'!';}
    }
}
```

Ce se va afișa la apelul $f(24)$?

- a. 24!6421!!!!
- b. 24*8421!!!!
- c. 24*8421**!!
- d. 24*8421!!!!!!

5. Se dă mulțimea fructelor {kiwi, portocale, banane, mere, prune, struguri}. Se vor forma platouri de fructe formate din câte 3 tipuri de astfel de fructe, folosind un algoritm backtracking, astfel încât să nu fie împreună pe același platou, mere cu struguri și nici banane cu prune. Care este penultima variantă de platou?

- a. banane, mere, prune
c. portocale, mere, banane

- b. banane, prune, struguri
d. portocale, mere, prune

Subiectul II (40p)

1. Algoritmul alăturat este reprezentat în pseudocod.

a) Care sunt valorile afișate în urma executării, dacă se citește succesiunea de valori: 5, 7, 8, 1, 0, 6? **(6p)**

b) Precizați o succesiune de 7 valori care pot fi citite astfel încât instrucțiunile din structura repetitivă *cât timp* să nu se execute niciodată. **(6p)**

c) Realizați programul C/C++ corespunzător algoritmului dat. **(10p)**

d) Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască fiecare structură repetitivă *pentru* cu câte o structură repetitivă condiționată anterior. **(6p)**

```
citește n (număr natural,
1<n<=100)
┌ pentru i←1, n execută
|   citește ai
└ ──┐
┌ pentru i ← 2, n execută
|   x←ai ; j← i-1
|   ┌ cât timp j>0 și x<aj execută
|   |   aj+1 ← aj
|   |   aj ← x
|   |   j ← j-1
|   └ ──┐
└ ──┐
┌ pentru i←1, n execută
|   scrie ai
└ ──┐
```

2. Se consideră declarațiile de mai jos, în care variabila *t* memorează simultan lungimile laturilor unui triunghi (numere reale pozitive). Fără a utiliza alte variabile suplimentare, scrieți o expresie C/C++ care are valoarea 1 dacă triunghiul ale cărui laturi sunt memorate în variabila *t* este triunghi isoscel, dar nu este echilateral.

```
struct triunghi { float a,b,c; } t;
```

(6p.)

3. 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. Fără a utiliza alte variabile, decât cele menționate, scrieți secvența de instrucțiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila *a* să memoreze tabloul alăturat. **(6p.)**

```
for(i=0;i<5;i++)
    for(j=0;j<5;j++)
        .....
5 4 4 4 4
4 5 2 1 0
2 2 5 2 2
4 3 2 5 0
0 0 0 0 5
```

**SUBIECTUL III (30p)**

1. Subprogramul **modifica** are doi parametri - v , prin care primește un tablou unidimensional cu maximum 100 de numere naturale mai mici decât 1000000 și n , numărul efectiv de elemente ale tabloului. Subprogramul înlocuiește fiecare element din tablou cu o valoare pozitivă care are exact același număr de cifre și toate cifrele sale sunt egale cu prima cifră pară nenulă a elementului (prima cifră pară nenulă de la stânga la dreapta). Dacă un element din tablou nu conține cifre pare nenule se va înlocui cu 0. Scrieți definiția completă a subprogramului.

Exemplu:

dacă $n=5$ și $v=(100, 425, 128, 7082, 35)$ atunci, după apel, vectorul va fi $v=(0, 444, 222, 8888, 0)$

(10p.)

2. O expresie are cel mult 20 de caractere și este formată din numere naturale cu cel mult cinci cifre, operatorii matematici: * (pentru înmulțire), + (pentru adunare), - (pentru scădere) și caracterul = (egal). Expresiile conțin un singur operator matematic între numere, încep cu un număr, se termină cu caracterul = (egal) și sunt scrise pe câte un rând fiecare, ca în exemplu.

Scrieți un program C/C++ care citește de la tastatură, un număr natural n ($0 < n < 20$) și apoi n expresii de tipul precizat mai sus și afișează pe ecran, pe câte un rând fiecare, cele n expresii evaluate corect matematic, ca șiruri de caractere.

Exemplu:

pentru $n=4$ și expresiile:

12*40=

1228+29=

1300-1=

7*11=

se afișează pe ecran:

12*40=480

1228+29=1257

1300-1=1299

7*11=77

(10p.)

3. Fișierul **numere.in** conține pe prima linie un număr natural n ($n \in [2, 10^9]$), iar pe a doua linie un șir de cel mult 10^9 numere naturale din intervalul $[1, n]$. Numerele din șir sunt ordonate crescător și sunt separate prin câte un spațiu.

Se cere să se determine valorile naturale distincte din intervalul $[1, n]$ care **NU** se găsesc în șirul menționat mai sus. Valorile determinate se afișează pe ecran în ordine strict crescătoare, separate prin câte un spațiu. Dacă nu există astfel de valori, se afișează pe ecran mesajul **Nu exista**.

Pentru determinarea valorilor cerute se utilizează un algoritm eficient din punctul de vedere al memoriei și al timpului de executare.

Exemplu: dacă fișierul conține numerele

10

3 4 4 8

se afișează pe ecran valorile

1 2 5 6 7 9 10

a. Descrie în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**

b. Scrie programul C/C++ corespunzător algoritmului proiectat. **(8p.)**