

**Examenul de bacalaureat național 2013**  
**Proba E. d)**  
**Informatică**  
**Limbajul C/C++**

**Varianta 2**

**Filiera teoretică, profilul real, specializările: matematică-informatică**

**matematică-informatică intensiv informatică**

**Filiera vocațională, profilul militar, specializarea matematică-informatică**

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 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**

**(30 de puncte)**

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

1. Variabila **x** este de tip întreg și poate memora un număr natural cu cel mult două cifre. Valoarea maximă pe care o poate avea expresia C/C++ alăturată este: **(4p.)** **x%4**

**a. 3**

**b. 24.75**

**c. 95**

**d. 396**

**2. Se consideră algoritmul alăturat, reprezentat în pseudocod.**

S-a notat cu **x%y** restul împărțirii numărului natural **x** la numărul natural nenul **y** și cu **[z]** partea întreagă a numărului real **z**.

**a) Scrieți valoarea afișată dacă se citește, în această ordine, numerele 19, 23 și 2. (6p.)**

**b) Dacă pentru variabila a se citește valoarea 1, iar pentru variabila c se citește valoarea 1, scrieți toate numerele naturale care pot fi citite pentru variabila b, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze valoarea 12. (4p.)**

citește **a,b,c** (numere naturale,  
**a ≤ b, 0 ≤ c ≤ 9**)

**s ← 0**

**pentru x ← a, b execută**

**y ← x**

**cât timp y > 0 execută**

**dacă y % 10 = c atunci**

**s ← s + 1**

**■**

**y ← [y / 10]**

**■**

**■**

**scrie s**

**c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască structura cât timp...execută cu o structură repetitivă cu test final. (6p.)**

**d) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)**

**SUBIECTUL al II-lea**

**(30 de puncte)**

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

1. Expresia `strlen("bine")` are valoarea: **(4p.)**

a. 1                      b. 4                      c. 5                      d. 6

2. Se consideră un graf orientat cu 6 vârfuri și fără circuite. Numărul maxim de arce ale grafului este: **(4p.)**

a. 5                      b. 7                      c. 10                      d. 15

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

3. Se consideră declararea alăturată. Scrieți o secvență de instrucțiuni în urma executării căreia câmpul `paritate` al variabilei `x` să memoreze litera `p`, dacă numărul memorat în câmpul `v` al acesteia este par, sau litera `i` dacă numărul memorat în câmpul `v` al acesteia este impar. **(6p.)**

```
struct numar  
{  
    int v;  
    char paritate;  
}  
x;
```

4. Se consideră un arbore cu 5 noduri, dintre care doar trei au gradul egal cu 1. Scrieți două valori care să reprezinte gradele celorlalte două noduri. **(6p.)**

5. Scrieți un program C/C++ care citește de la tastatură două numere naturale `m` și `n` ( $2 \leq m \leq 50$ ,  $2 \leq n \leq 50$ ) și elementele unui tablou bidimensional cu `m` linii și `n` coloane, numere naturale cu cel mult patru cifre fiecare. Programul determină coloanele care au toate elementele egale cu aceeași valoare și, pentru fiecare astfel de coloană afișează pe ecran valoarea respectivă. Valorile afișate sunt separate prin câte un spațiu, iar dacă nu există astfel de valori, programul afișează pe ecran mesajul `Nu exista`.

**Exemplu:** pentru `m=4`, `n=5` și tabloul alăturat, se afișează pe ecran numerele 3 7 7, nu neapărat în această ordine.

**(10p.)**

2	3	7	5	7
3	3	7	4	7
7	3	7	1	7
8	3	7	6	7

**SUBIECTUL al III-lea**

**(30 de puncte)**

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

1. Se consideră subprogramele **f1** și **f2**, definite mai jos.

```
int f1 (int x, int y)
{
    return x*y;
}
```

```
int f2 (int x, int y)
{
    if (y==1) return x;
    else return x*f2(x, y-1);
}
```

La apel, pentru parametrii  $x=2$  și  $y=3$ , returnează  $x^y$ :

**(4p.)**

- a. atât **f1**, cât și **f2**      b. numai **f1**      c. numai **f2**      d. nici **f1**, nici **f2**

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

2. Utilizând metoda backtracking se generează toate cuvintele de 5 litere din mulțimea  $\{A, M\}$ , cuvinte care încep și se termină cu **M**. Primele patru soluții generate sunt, în această ordine: **MAAAM**, **MAAMM**, **MAMAM**, **MAMMM**. Scrieți cea de a cincea și cea de a șasea soluție, în ordinea generării acestora. **(6p.)**

3. Se consideră subprogramul **valuri**, cu doi parametri:

- **n**, prin care primește o valoare naturală  $2 < n < 50$ ;
- **v**, prin care furnizează un tablou unidimensional cu  $2 \cdot n$  elemente, valori întregi din intervalul  $[1, 2 \cdot n]$ .

Subprogramul construiește tabloul **v** astfel încât, în acesta, șirul elementelor impare să fie strict crescător, iar șirul elementelor pare să fie strict descrescător. Primul element al tabloului este impar, iar două elemente cu aceeași paritate nu pot ocupa poziții consecutive în tablou, ca în exemplu.

Scrieți definiția completă a subprogramului.

**Exemplu:** dacă  $n=4$  atunci, după apel,  $v = (1, 8, 3, 6, 5, 4, 7, 2)$ .

**(10p.)**

4. Se consideră șirul definit alăturat (unde **n** este un număr natural nenul), în care nu există doi termeni cu aceeași paritate aflați pe poziții consecutive:

1, 2, 3, 4, 7, 8, 15, 16 . . . .

$$f_n = \begin{cases} 1, & \text{dacă } n = 1 \\ 1 + f_{n-1}, & \text{dacă } n \text{ par} \\ 1 + 2 \cdot f_{n-2}, & \text{altfel} \end{cases}$$

Se citește de la tastatură un număr natural **x**, cu cel mult nouă cifre, termen al șirului dat, și se cere să se scrie în fișierul text **bac.txt**, în ordine strict descrescătoare, separați prin câte un spațiu, toți termenii șirului care sunt mai mici sau egali cu **x**.

Pentru determinarea termenilor ceruți se utilizează un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

**Exemplu:** dacă  $x=15$ , fișierul **bac.txt** conține numerele

15 8 7 4 3 2 1

- a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**

- b) Scrieți programul C/C++ corespunzător algoritmului descris. **(6p.)**