

EXAMENUL DE BACALAUREAT – 2009
Proba scrisă la INFORMATICĂ
PROBA E, limbajul C/C++
Specializarea Matematică-informatică intensiv informatică

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru 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).

Subiectul I (30 de puncte)

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

1. Cărui interval îi aparține valoarea memorată de variabila reală x astfel încât expresia următoare, scrisă în limbajul **C/C++**, să aibă valoarea 1? (4p.)
- $(x \leq -2) \ || \ (x > -1) \ \&\& \ !(x \geq 1) \ || \ (x > 50)$**
- a. $(-\infty, -2] \cup (-1, 1) \cup (50, \infty)$ b. $(-\infty, -2) \cup (-1, 50)$
c. $(-\infty, -2) \cup (-1, 1] \cup (50, \infty)$ d. $(-\infty, -2) \cup (-1, 1) \cup (50, \infty)$

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

2. Se consideră algoritmul alăturat, descris în pseudocod.

S-a notat cu $x \% y$ restul împărțirii numărului întreg x la numărul întreg nenul y și cu $[a]$ partea întreagă a numărului real a .

- a) Scrieți care este valoarea afișată dacă se citesc, în această ordine, numerele **5, 12, 4, 13, 25, 17**. (6p.)
- b) Scrieți un șir de date de intrare ce pot fi citite astfel încât valoarea afișată să fie **4**. (4p.)
- c) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura repetitivă **pentru... execută** cu o structură repetitivă cu test final. (6p.)
- d) Scrieți programul **C/C++** corespunzător algoritmului dat. (10p.)

```
citește n
      (număr natural nenul)
d ← 0
c ← 0
pentru i ← 1, n execută
    citește x
      (număr natural nenul)
    cât timp x%2=0 execută
        x ← [x/2]; d ← d+1
    cât timp x%5=0 execută
        x ← [x/5]; c ← c+1
dacă c < d atunci
    scrie c
altfel
    scrie d
```

Subiectul II (30 de puncte)

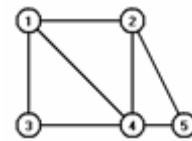
Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.

1. Considerăm declararea alăturată folosită pentru a memora numele, prenumele și cele 2 note ale unui elev.
Care dintre instrucțiunile de mai jos calculează în variabila reală m media aritmetică a notelor elevului ale cărui informații sunt memorate în variabila x ? (4p.)

```
struct elev{  
    char nume[10], prenume[20];  
    float nota1, nota2;  
} x;
```

- a. $m=(x.nota1+x.nota2)/2;$ b. $m=(nota1+nota2)/2;$
c. $x.m=(x.nota1+x.nota2)/2;$ d. $m=(x, nota1+x, nota2)/2;$

2. Se consideră graful neorientat din figura alăturată. Care este numărul **minim** de muchii ce se pot elimina astfel încât graful parțial obținut să aibă exact 3 componente conexe? (4p.)



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

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

3. În secvența alăturată, variabila a memorează elementele unui tablou bidimensional cu 4 linii (numerotate de la 0 la 3) și 4 coloane (numerotate de la 0 la 3), iar toate celelalte variabile sunt de tip întreg.
Ce valoare va avea elementul $a[3][3]$ și care este suma elementelor de pe prima linie a tabloului după executarea secvenței de instrucțiuni scrisă alăturat? (6p.)

```
x=5;  
for (i=0; i<=3; i++)  
    for (j=0; j<=3; j++)  
        if (i==j)  
            {a[i][j]=x;  
             x=x+1;}
```

4. Se consideră o listă liniară simplu înlănțuită, alocată dinamic, cu cel puțin 5 noduri, în care fiecare nod al listei conține în câmpul **urm** adresa nodului următor din listă sau **NULL** dacă nu există un nod următor. Adresa primului nod este memorată în variabila **prim**, iar variabila **p** este de același tip cu **prim**.

Ce modificare se produce asupra listei prin executarea secvenței de instrucțiuni alăturate? (6p.)

```
p=prim;  
prim=p->urm;  
delete(p);    |    free(p);
```

5. Scrieți un program **C/C++** care citește de la tastatură un număr natural nenul n ($n \leq 100$), apoi n șiruri de caractere, fiecare șir având maximum 30 de caractere, doar litere mici ale alfabetului englez, și afișează pe ecran câte dintre cele n șiruri de caractere sunt formate **numai** din vocale. Se consideră vocale literele **a, e, i, o, u**.

Exemplu: dacă $n=3$, iar șirurile citite sunt date alăturat, atunci programul va afișa pe ecran **1** deoarece șirul **ae** este format numai din vocale. (10p.)

```
arc  
ae  
creion
```

Subiectul III (30 de puncte)

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

1. Utilizând metoda backtracking, se generează în ordine lexicografică toate anagramele cuvântului **caiet** (cuvinte formate din aceleași litere, eventual în altă ordine). Câte cuvinte care încep cu litera **t** vor fi generate? (4p.)
- a. 1 b. 6 c. 12 d. 24

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

2. Se consideră subprogramul **f**, definit alături. Ce valoare are **f(4)**? Dar **f(9)**? (6p.)
- | | |
|---|--|
| <pre>int f(int n) { if (n==0) return 0; if(n%2==1) return n-f(n-1); return f(n-1)-n }</pre> | |
|---|--|
3. Subprogramul **mult**, cu doi parametri, primește prin intermediul primului parametru, **n**, un număr natural nenul cu maximum trei cifre și prin intermediul celui de-al doilea parametru, **a**, un tablou unidimensional având **n** componente numere naturale cu cel mult **8** cifre fiecare. Subprogramul returnează valoarea **true** dacă cele **n** componente ale lui **a** pot forma o mulțime și returnează **false** în caz contrar.
- a) Scrieți definiția completă a subprogramului **mult**. (6p.)
- b) Fișierul text **date.in** conține cel mult **400** de numere naturale având maximum **8** cifre fiecare. Scrieți un program **C/C++** care, folosind apeluri utile ale subprogramului **mult**, afișează pe ecran valoarea maximă **k**, astfel încât primele **k** numere succesive din fișier să poată forma o mulțime.
- Exemplu:** dacă fișierul **date.in** conține
- 16 17 8 31 8 2 10**
- atunci se va afișa **4** (deoarece primele patru numere din fișier pot forma o mulțime și acesta este cardinalul maxim posibil în condițiile impuse de enunțul problemei) (4p.)
4. Scrieți un program **C/C++** care citește de la tastatură două numere naturale nenule **n** și **k** (**k < n ≤ 10000**) și afișează pe ecran un șir format din **k** numere naturale consecutive care au suma **n**. Numerele se vor afișa pe ecran, în ordine crescătoare, despărțite între ele prin câte un spațiu. În cazul în care nu există un astfel de șir, se va afișa mesajul **Nu există**. (10p.)