

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. Se consideră variabila **a** care memorează un număr cu exact 6 cifre. Care dintre expresiile C/C++ de mai jos are ca valoare numărul format din cele două cifre din mijloc ale valorii memorate în **a**? **(4p.)**
- a. $(a \% 100) / 100$ b. $a / 100 \% 100$
c. $a / 1000 + a \% 1000$ d. $a / 100 \% 10 + a / 1000 \% 10$

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

2. Se consideră algoritmul alăturat, descris în pseudocod:

S-a notat cu $[x]$, partea întreagă a numărului real x .

- a) Scrieți valorile care se vor afișa pentru **a=9**. **(4p.)**

- b) Scrieți numărul valorilor din intervalul $[1, 5]$ care, citite pentru variabila **a**, determină, după executarea algoritmului alăturat, memorarea valorii 1 în variabila **b**. **(6p.)**

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

- d) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să nu se utilizeze structuri repetitive sau subprograme recursive. **(6p.)**

```
citește a
(număr natural, a>0)
k ← 0
b ← [(a+1)*(a+2)/2]
cât timp b ≥ a execută
|   b ← b - a
|   k ← k + 1
|   ■
scrie b, k
```

Subiectul II (30 de puncte)

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

1. Se consideră graful neorientat cu 7 noduri, numerotate de la 1 la 7, și muchiile [1,3], [2,3], [3,4], [3,5], [5,4], [1,2], [2,5], [2,4], [6,7], [3,6]. Care dintre următoarele succesiuni de noduri reprezintă un lanț care trece o singură dată prin toate nodurile grafului? **(4p.)**
 - a. (1, 2, 3, 4, 5, 6, 7)
 - b. (4, 5, 3, 6, 7)
 - c. (7, 6, 3, 5, 4, 2, 1)
 - d. (1, 3, 5, 4, 2, 3, 6)
2. Un arbore cu 11 noduri, numerotate de la 1 la 11, este memorat cu ajutorul vectorului de „tați” $t = (2, 5, 5, 3, 0, 2, 4, 6, 6, 2, 3)$. Mulțimea tuturor ascendenților nodului 8 este: **(4p.)**
 - a. {1, 2, 5, 6, 10}
 - b. {6, 2, 5}
 - c. {6}
 - d. {5, 2}

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

3. Scrieți definiția corectă a unui tip de date necesar pentru a memora simultan, într-o singură variabilă de acest tip, următoarele caracteristici ale unui autoturism: marca (cuvânt de maximum 20 caractere) și anul fabricației (număr natural format din exact 4 cifre), astfel încât expresia C/C++ de mai jos să aibă ca valoare vechimea mașinii ale cărei caracteristici sunt memorate în variabila **x**.
2008-x.anul_fabricatiei **(6p.)**
4. În secvența alăturată, variabilele **i** și **j** sunt de tip întreg, iar variabila **a** memorează o matrice în care prima linie și prima coloană sunt numerotate cu 1. **Toate** elementele matricei primesc valori în urma executării secvenței. Scrieți în ordine, începând cu prima linie, doar elementele situate pe cea de-a treia coloană a matricei. **(6p.)**

for(j=1;j<=6;j++)
for(i=1;i<=4;i++)
a[i][j]=2*i+j;

for(j=1;j<=6;j++)
for(i=1;i<=4;i++)
a[i][j]=2*i+j;
5. Se consideră o listă liniară simplu înlănțuită alocată dinamic în care fiecare nod memorează în câmpul **info** un cuvânt format din cel mult 20 de caractere, doar litere mari ale alfabetului englez, iar în câmpul **urm**, adresa următorului nod al listei sau **NULL** dacă nu există un element următor. Scrieți un program C/C++ care citește de la tastatură un număr natural **n** ($n \leq 100$) și apoi **n** cuvinte distincte, fiecare cuvânt fiind format din cel mult 20 de caractere, doar litere mari ale alfabetului englez, și construiește o listă simplu înlănțuită, cu acele cuvinte citite, care încep și se termină cu aceeași literă. Cuvintele se vor memora în listă în ordine inversă citirii lor. Programul va afișa pe ecran cuvintele din listă, în linie, separate prin câte un spațiu.

Exemplu: pentru **n=9** și cuvintele citite:

URSUZU IRI E SUPARAT POP DORIS SI ANA CITESC

lista va fi:



(10p.)

Subiectul III (30 de puncte)

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

1. Având la dispoziție cifrele 0, 1 și 2 putem genera, în ordine crescătoare, numere care au suma cifrelor egală cu 2 astfel încât primele 6 numere generate sunt, în această ordine: 2, 11, 20, 101, 110, 200. Folosind același algoritm se generează numere cu cifrele 0, 1, 2 și 3 care au suma cifrelor egală cu 4. Care va fi al 7-lea număr din această generare ? **(4p.)**
- a. 103 b. 301 c. 220 d. 130

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

2. Se consideră subprogramul recursiv `f1` definit alăturat. Ce se va afișa în urma apelului `f1(5);`? **(6p.)**
- ```
void f1(int x)
{ if (x<=9)
 { cout<<x+1; | printf("%d",x+1);
 f1(x+2);
 cout<<x+3; | printf("%d",x+3);
 }
}
```
3. Scrieți definiția completă a subprogramului `suma` care primește ca parametru un tablou unidimensional `x` cu cel mult 100 de elemente, numere reale, un număr natural `n` ce reprezintă numărul efectiv de elemente ale tabloului `x` ( $n \leq 100$ ), și un număr natural `m` ( $n \geq m$ ). Subprogramul returnează suma obținută din cele mai mici `m` elemente ale tabloului `x`. **(10p.)**
4. În fișierul `numere.txt` se află memorate, pe prima linie un număr natural `n` ( $1 \leq n \leq 100$ ), iar pe fiecare dintre următoarele `n` linii, câte două numere întregi `x, y` ( $-100 \leq x \leq y \leq 100$ ), reprezentând capetele câte unui segment `[x, y]` desenat pe axa `Ox` de coordonate.
- a) Scrieți în limbajul C/C++ un program eficient din punct de vedere al timpului de executare și al spațiului de memorare, care citește din fișier datele existente, determină segmentul rezultat în urma intersecției tuturor celor `n` segmente date și afișează pe ecran două numere despărțite printr-un spațiu ce reprezintă capetele segmentului cerut. Dacă segmentele nu au nici un punct comun se va afișa pe ecran valoarea 0. **(6p.)**
- b) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**

**Exemplu:** dacă fișierul `numere.txt` are conținutul alăturat, se va afișa

|          |       |
|----------|-------|
| pe ecran | 5     |
| 3 5      | -7 10 |
|          | 3 20  |
|          | -5 5  |
|          | 0 12  |
|          | -8 30 |