

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. Care dintre următoarele instrucțiuni C/C++ determină eliminarea cifrei din mijloc a unui număr natural, cu exact 5 cifre, memorat în variabila x? (4p.)
- a. $x = x / 1000 * 100 + x \% 100;$
- b. $x = x \% 1000 * 100 + x / 100;$
- c. $x = x / 100 * 100 + x \% 100;$
- d. $x = x / 1000 + x \% 100;$

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

- 2. Se consideră algoritmul alăturat, descris în pseudocod.**

S-a notat cu $a \% b$ restul împărțirii numărului natural a la numărul natural, nenul, b și cu $a \leftrightarrow b$ interschimbarea valorilor reținute de variabilele a și b .

- a) Scrieți succesiunea de caractere care se vor afișa în urma executării algoritmului dacă se citesc, în această ordine, valorile 2 și 9. (6p.)

- b)** Știind că pentru variabila y se citește valoarea 79, scrieți toate valorile distincte care pot fi citite pentru variabila x , astfel încât să fie afișat de exact 40 de ori caracterul *.

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

- d) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura repetitivă **cât timp...execută** cu o structură repetitivă de alt tip. **(6p.)**

```

citește x,y
    (numere naturale)
dacă x>y atunci
    y←→x
    ■
dacă x%2=0 atunci
    x←x+1
    ■
cât timp x≤y execută
    x←x+2
    scrie ``'
    ■

```

Subiectul II (30 de puncte)

Pentru fiecare dintre itemii următori, scrieți pe foaia de examen litera care corespunde răspunsului corect.

1. Fiecare element al unei liste circulare, simplu înălțuită, nevidă, alocată dinamic, memorează în câmpul `val` o valoare întreagă, iar în câmpul `adr` adresa elementului următor din listă. Știind că variabila `p` reține adresa unui element oarecare din listă, iar variabila `q` este de același tip cu `p`, precizați care dintre următoarele secvențe de program afișează, în urma executării, toate valorile memorate de elementele listei? **(4p.)**

- | | |
|--|--|
| <p>a. <code>q=p;</code>
<code>while(q!=p) {cout<<q->val;</code>
<code> q=q->adr;}</code></p> <p>b. <code>q=p;</code>
<code>while(q->adr!=p){cout<<q->val;</code>
<code> q=q->adr;}</code></p> <p>c. <code>q=p;</code>
<code>do{ cout<<q->val;</code>
<code> q=q->adr;</code>
<code> }while(q!=p);</code></p> <p>d. <code>q=p->adr;</code>
<code>while(q!=p) {cout<<q->val;</code>
<code> q=q->adr;}</code></p> | <p><code>q=p;</code>
<code>while(q!=p) { printf("%d",q->val);</code>
<code> q=q->adr; }</code></p> <p><code>q=p;</code>
<code>while(q->adr!=p){ printf("%d",q->val);</code>
<code> q=q->adr; }</code></p> <p><code>q=p;</code>
<code>do{ printf("%d",q->val);</code>
<code> q=q->adr;</code>
<code> }while(q!=p);</code></p> <p><code>q=p->adr;</code>
<code>while(q!=p) { printf("%d",q->val);</code>
<code> q=q->adr; }</code></p> |
|--|--|

2. Care dintre următoarele variante reprezintă o declarație corectă pentru o variabilă `x` care memorează simultan coordonatele reale (abscisa și ordonata) ale unui punct în planul `xOy`? **(4p.)**

- | | |
|---|----------------------------|
| a. <code>struct punct{float ox,oy;} x;</code> | b. <code>char x[2];</code> |
| c. <code>struct x{float ox,oy;};</code> | d. <code>float x;</code> |

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

3. Un arbore binar este un arbore cu rădăcină în care fiecare nod are cel mult 2 descendenți direcți (fii). Înălțimea unui arbore este reprezentată de numărul maxim de muchii ale unui lanț elementar ce unește rădăcina cu un vârf terminal (frunză). Pentru un arbore binar cu exact 8 noduri, care este înălțimea minimă posibilă și care poate fi numărul maxim de noduri terminale (frunze) ale arborelui în acest caz? **(6p.)**

4. În secvența de program alăturată, variabila `a` memorează un șir cu cel mult 100 de caractere, iar variabila `i` este de tip întreg. Completați punctele de suspensie din secvență astfel încât, în urma executării secvenței, aceasta să afișeze șirul de caractere `*nf*rm*t*c*`. **(6p.)**

```
strcpy(a,"informatica");
for(i=0;i<strlen(a);i++)
    if(...)
        cout<<...; | printf(...);
    else
        cout<<...; | printf(...);
```

5. Scrieți un program C/C++ care citește de la tastatură un număr natural `n` ($2 < n < 20$), construiește în memorie și afișează pe ecran o matrice cu `n` linii și `n` coloane, numerotate de la 1 la `n` în care fiecare element din matrice aflat pe o linie impară va fi egal cu numărul liniei pe care se află și fiecare element aflat pe o linie pară va fi egal cu numărul coloanei pe care se află.

Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

Exemplu: pentru `n=5` se va afișa matricea alăturată.

(10p.)

1	1	1	1	1
1	2	3	4	5
3	3	3	3	3
1	2	3	4	5
5	5	5	5	5

Subiectul III (30 de puncte)

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

1. Funcția `f` are definiția alăturată. Ce se va afișa în urma apelului `f(12345,0);`? **(4p.)**

```
void f(long n, int i)
{
    if (i < n % 10)
    {
        cout << n % 10; | printf("%d", n % 10);
        f(n / 10, i + 1);
    }
}
```

a. 54321

b. 543

c. 54

d. 5432

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare

2. Un algoritm generează, în ordine lexicografică, toate șirurile alcătuite din câte n cifre binare (0 și 1). Știind că pentru $n=5$, primele patru soluții generate sunt 00000, 00001, 00010, 00011, precizați care sunt ultimele **trei** soluții generate, în ordinea obținerii lor. **(6p.)**
3. Scrieți definiția completă a subprogramului `count` care are doi parametri, a și n , prin care primește un tablou unidimensional cu maximum 100 de numere reale și respectiv numărul efectiv de elemente din tablou. Subprogramul returnează numărul de elemente din tabloul a care sunt mai mari sau cel puțin egale cu media aritmetică a tuturor elementelor din tablou.
Exemplu: dacă tabloul are 6 elemente și este de forma (12, 7.5, 6.5, 3, 8.5, 7.5), subprogramul va returna valoarea 4 (deoarece media tuturor elementelor este 7.5 și numerele subliniate sunt cel puțin egale cu această medie). **(10p.)**
4. În fișierul `numere.txt` este memorat un șir de maximum 10000 numere naturale, distincte două câte două, cu maximum 4 cifre fiecare, separate prin câte un spațiu. Pentru un număr k citit de la tastatură, se cere afișarea pe ecran a poziției pe care se va găsi acesta în șirul de numere din fișier, dacă șirul ar fi ordonat descrescător, sau mesajul **nu există**, dacă numărul k nu se află printre numerele din fișier. Alegeți un algoritm eficient de rezolvare din punct de vedere al memoriei utilizate și al timpului de executare.
Exemplu: dacă fișierul `numere.txt` conține numerele 26 2 5 30 13 45 62 7 79, iar k are valoarea 13, se va afișa 6 deoarece 13 s-ar găsi pe poziția a șasea în șirul ordonat descrescător (79 62 45 30 26 13 7 5 2).
a) Descrieți succint, în limbaj natural, strategia de rezolvare și justificați eficiența algoritmului ales. **(4p.)**
b) Scrieți programul C/C++ corespunzător algoritmului ales. **(6p.)**