

EXAMENUL DE BACALAUREAT – 2008
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 expresii C/C++ are valoarea 1 dacă variabila **x** memorează un număr natural pătrat perfect? (4p.)
- a. `sqrt(x) == floor(sqrt(x))` b. `sqrt(x) != floor(sqrt(x))`
c. `floor(sqrt(x)) == ceil(sqrt(x))` d. `floor(sqrt(x)) == ceil(sqrt(x))`

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 natural ***x*** la numărul natural nenul ***y***, iar cu ***[x/y]*** câtul împărțirii întregi a numărului natural ***x*** la numărul natural nenul ***y***.

- a) Scrieți ce se va afișa dacă se citesc pentru **a** valoarea **625231**, iar pentru **b** valoarea **425271**. (6p.)
- b) Dacă **a=12345** scrieți o valoare de cinci cifre care poate fi citită pentru **b**, astfel încât să se afișeze pentru **p** valoarea **42**. (6p.)

```
citește a,b (numere naturale)
c ← 0
p ← 0
cât timp a + b > 10 execută
    dacă (a%10 = b%10) și (a%10%2=1)
        atunci c ← c*10 + b%10
        altfel p ← p*10 + a%10
    a ← [a/10]
    b ← [b/10]
scrie c, p
```

- c) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d) Efectuați un număr minim de modificări asupra algoritmului dat astfel încât prin executarea algoritmul obținut să se calculeze în variabila **c** numărul de cifre impare, comune și aflate pe poziții identice, pe care le au cele două valori citite. (4p.)

Subiectul II (30 de puncte)

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

1. Într-o listă liniară simplu înlănțuită, cu cel puțin 3 noduri, fiecare element reține în câmpul **urm** adresa următorului element din listă, iar în câmpul **info** informația utilă de tip întreg. Dacă variabila **p** reține adresa primului element din listă atunci care dintre secvențele de mai jos atribuie câmpului **info** al celui de al treilea nod informația utilă din primul nod al listei? (4p.)
- a. **p->urm->urm->info=p->info;** b. **p->urm->urm->info=p->urm->info;**
c. **p->info->info->info = p->info;** d. **p->urm->urm = p->info;**
2. Ce se va afișa în urma executării secvenței de program de mai jos dacă variabila **x** memorează cuvântul **bacalaureat**, iar variabila **y** memorează cuvântul **banal**? (4p.)
- ```
if(strcmp(x, y) > 0) cout << x; | printf("%s", x);
else
 if(strcmp(x,y) < 0) cout << y; | printf("%s", y);
 else cout << "imposibil"; | printf("imposibil");
```
- a. **imposibil**                                              b. **bacalaureat**  
c. **banal**                                                  d. **bacalaureatimposibil**

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre întrebările următoare:**

3. Se consideră un arbore cu 9 noduri, numerotate de la 1 la 9, și cu vectorul "de tați" următor: (8, 8, 8, 2, 6, 2, 9, 0, 2). (3p.)
- a) Enumerați descendenții nodului 2. (3p.)  
b) Câte noduri de tip frunză are acest arbore? (3p.)
4. Se consideră graful neorientat cu 6 noduri, numerotate de la 1 la 6 și următoarele muchii: [1, 3] [1, 5] [2, 3] [2, 4] [2, 6] [5, 3] [6, 4]. (3p.)
- a) Care este numărul minim de muchii ce trebuie eliminate din acest graf, astfel încât graful parțial obținut să nu conțină niciun ciclu? (3p.)  
b) Care este numărul minim de muchii ce trebuie eliminate din acest graf, astfel încât graful parțial obținut să aibă exact două componente conexe? (3p.)
5. Se consideră șirul lui Fibonacci, definit astfel:  $f_0 = 0$ ,  $f_1 = 1$ ,  $f_n = f_{n-1} + f_{n-2}$ , dacă  $n > 1$ . Scrieți un program C/C++ care citește de la tastatură o valoare naturală **n** ( $2 \leq n \leq 50$ ), construiește în memorie și afișează pe ecran o matrice **a** cu **n** linii și **n** coloane ale cărei elemente sunt numere naturale, fiecare reprezentând ultima cifră a câte unui termen al șirului lui Fibonacci, începând de la termenul de indice 1 și până la termenul de indice  $n^2$ , în ordine, linie cu linie de sus în jos, și în cadrul fiecărei linii de la stânga la dreapta, ca în exemplu. (10p.)
- Matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate între ele prin câte un spațiu.
- Exemplu:** pentru **n = 4** se va obține matricea alăturată.
- |   |   |   |   |
|---|---|---|---|
| 1 | 1 | 2 | 3 |
| 5 | 8 | 3 | 1 |
| 4 | 5 | 9 | 4 |
| 3 | 7 | 0 | 7 |

**Subiectul III (30 de puncte)**

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

1. Prin metoda backtracking se generează toate anagramele (cuvintele obținute prin permutarea literelor) unui cuvânt dat. Știind că se aplică această metodă pentru cuvântul **pescar**, precizați câte cuvinte se vor genera astfel încât prima și ultima literă din fiecare cuvânt generat să fie **vocală** (sunt considerate vocale caracterele **a, e, i, o, u**)? **(4p.)**
- a. 96                      b. 24                      c. 48                      d. 12

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

2. Se consideră funcția **Suma**, definită alături. Ce valoare are **Suma(8)**? Dar **Suma(11)**? **(6p.)**
- ```
int Suma(int x)
{ if(x == 1) return 0;
  if(x%2==0) return Suma(x-1) + (x-1)*x;
  return Suma(x-1) - (x-1)*x;
}
```
3. Scrieți definiția completă a subprogramului **Cifra** cu doi parametri **n** și **x**, care primește prin intermediul parametrului **n** un număr natural de cel mult nouă cifre și furnizează prin parametrul **x** cea mai mare cifră a numărului transmis prin parametrul **n**. **(10p.)**
4. a) Se numește “număr mare” un număr care are mai mult de nouă cifre. Scrieți un program **C/C++** care citește din fișierul text **NUMERE.IN** un număr natural **n** (**10 < n < 1000**) apoi de pe următoarea linie **n** numere naturale cu cel mult nouă cifre fiecare, dintre care cel puțin unul nenul, și folosind apeluri ale funcției **Cifra** definită la punctul 3 construiește în mod eficient din punct de vedere al memoriei folosite, cel mai mic “număr mare” cu exact **n** cifre. Fiecare dintre cifrele numărului mare reprezintă cifra de valoare maximă a unui număr citit din fișier și nu există două cifre în “numărul mare” care să aparțină aceluiași număr citit. Scrieți în fișierul text **NUMERE.OUT** “numărul mare” obținut astfel. **(6p.)**
- b) Descrieți succint în limbaj natural metoda de rezolvare folosită explicând în ce constă eficiența ei (3 – 4 rânduri) **(4p.)**

Exemplu:

NUMERE.IN

10

725 3695 423 0 7895 0 100 101 870 568

NUMERE.OUT

1001478899