

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. Variabila reală **x** memorează numărul 2.682. Care dintre următoarele instrucțiuni **C/C++** va atribui variabilei reale **y** valoarea 10? (4p.)
- a. **y = (x*100)/100 + (x*1000%100/10);**
 - b. **y = floor(x*100)/100 + floor(x*100%100/10);**
 - c. **y = floor(x*100)/100 + floor(x*1000)%100/10;**
 - d. **y = floor(x*1000)/100 + floor(x*100%100/10);**

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

2. Se consideră algoritmul alăturat, descris în pseudocod.

S-a notat cu **x%y** restul împărțirii numărului natural **x** la numărul natural nenul **y**, iar prin **[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 260, pentru **b** valoarea 288 și pentru **p** valoarea 9. (6p.)
- b) Dacă **a=110**, iar **p=18**, scrieți cel mai mare număr care poate fi citit pentru **b**, astfel încât să se afișeze valoarea 0. (4p.)

```
citește a, b, p
(numere naturale nenule, a < b)
nr ← 0
pentru i ← a, b execută
    x ← i
    cât timp x ≠ 0 și x%p ≠ 0 execută
        x ← [x/10]
    dacă x ≠ 0 atunci
        nr ← nr + 1
scrie nr
```

- 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 **cât timp...execută** cu o structură repetitivă de un alt tip. (6p.)

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 arborele cu 12 noduri, numerotate de la 1 la 12, definit prin următorul vector „de tați”: (4, 8, 0, 3, 10, 1, 8, 3, 2, 4, 7, 10). Care dintre nodurile arborelui au exact un descendent direct (fiu)? (4p.)
- a. 6, 9, 11
b. 1, 2, 7
c. 5, 12, 6, 9, 11
d. 10, 1, 2, 7
2. Se consideră declarațiile alăturate. Care este tipul expresiei de mai jos? (4p.)
- | | | |
|-----------------------|-------------------------|----------------------------|
| <code>struct A</code> | <code>struct B</code> | <code>struct C</code> |
| <code>{ int x;</code> | <code>{ float x;</code> | <code>{ struct A x;</code> |
| <code>char y;</code> | <code>long y;</code> | <code>struct B y;</code> |
| <code>};</code> | <code>};</code> | <code>} x, y;</code> |
- `x.x.y`
- a. long b. int c. char d. float

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

3. Se consideră graful orientat cu 6 noduri, numerotate de la 1 la 6, și arcele (1, 2), (1, 5), (1, 6), (2, 3), (4, 3), (4, 5), (6, 5). Care este numărul minim de arce care trebuie adăugate grafului astfel încât acesta să conțină cel puțin un circuit elementar de lungime 4? Pentru graful rezultat, dați un exemplu de astfel de circuit. (6p.)
4. Variabilele **n**, **i**, **p** și **q** sunt de tip întreg, iar variabila **a** memorează un tablou bidimensional cu **n** linii și **n** coloane numerotate de la 1 la **n** ($0 < n < 50$), cu elemente numere reale. Cu ce instrucțiuni se poate înlocui zona punctată din secvența alăturată, astfel încât, în urma executării acesteia, să se interschimbe elementele liniei **q** cu elementele liniei **p** ale tabloului **a** ($1 \leq q \leq n, 1 \leq p \leq n$). Dacă sunt necesare și alte variabile, scrieți declarațiile acestora. (6p.)
- ```
for(i = 0; i < n; i++)
{.....}
```
5. Scrieți un program C/C++ care citește de la tastatură un număr natural **n** ( $0 < n < 101$ ) și apoi **n** propoziții. Fiecare propoziție este formată din maximum 255 de caractere care sunt numai litere mici ale alfabetului englez și spații. Știind că fiecare propoziție se termină cu caracterul **Enter**, să se afișeze pe ecran propoziția care are cele mai puține vocale (sunt considerate vocale caracterele **a**, **e**, **i**, **o**, **u**). Dacă două sau mai multe propoziții au același număr de vocale, se va afișa ultima propoziție, în ordinea citirii, care îndeplinește condiția. Exemplu: pentru **n = 4** și următoarele propoziții:
- ```
azi a plouat tare  
a fost innorat peste tot  
maine va fi soare  
acum nu mai vreau sa ploua  
se va afișa: maine va fi soare
```
- (10p.)

Subiectul III (30 de puncte)

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

1. Se generează în ordine crescătoare toate numerele de 4 cifre, cu cifre distincte, astfel încât diferența în valoare absolută dintre ultimele două cifre ale fiecărui număr generat este egală cu 2. Primele opt soluții generate sunt, în ordine: 1024, 1035, 1042, 1046, 1053, 1057, 1064, 1068. Care dintre următoarele numere se va genera imediat după numărul 8975? (4p.)
- a. 8979 b. 9013 c. 8957 d. 9024

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

2. Ce se va afișa în urma executării subprogramului alăturat, la apelul `F(56)`? (6p.)
- ```
void F(int x)
{
 if(x)
 {
 F(x/2);
 cout << x%10; | printf("%d", x%10);
 }
}
```
3. a) Scrieți definiția completă a subprogramului **Cautare**, cu 3 parametri, care primește prin parametrul **n** un număr natural ( $1 \leq n \leq 100$ ), prin parametrul **x** un tablou unidimensional format din **n** componente (numere întregi de cel mult patru cifre fiecare:  $x_1, x_2, \dots, x_n$ ) memorate în ordine crescătoare și prin parametrul **val** un număr întreg de cel mult patru cifre, diferit de oricare dintre elementele tabloului unidimensional **x**. Subprogramul va căuta, în mod eficient din punct de vedere al timpului de executare, poziția pe care ar trebui inserată valoarea **val** în șirul **x** astfel încât să se obțină tot un șir ordonat și returnează această poziție. (6p.)
- b) Descrieți succint, în limbaj natural, metoda de rezolvare folosită, explicând în ce constă eficiența ei (3 – 4 rânduri) (4p.)
4. Fișierul text **sir.in** conține cel mult 1000 numere naturale de maximum patru cifre fiecare, numerele fiind diferite două câte două și despărțite prin câte un spațiu. Scrieți un program C/C++ care citește numerele din fișierul **sir.in** și, folosind apeluri utile ale subprogramului **Cautare**, definit la punctul 3, construiește în memorie un tablou unidimensional care va conține toate numerele din fișierul **sir.in** ordonate crescător. Programul scrie în fișierul text **sir.out** șirul obținut, câte 10 elemente pe un rând, elementele de pe același rând fiind despărțite printr-un singur spațiu. **Exemplu:** dacă fișierul **sir.in** conține numerele: 7 -5 635 -456 0 8 587 -98 65 3 -8 atunci după executarea programului fișierul **sir.out** va conține:  
-456 -98 -8 -5 0 3 7 8 65 587  
635 (10p.)