

Examenul de bacalaureat 2010

Proba E-d)

Proba scrisă la INFORMATICĂ

Limbajul C/C++

Specializarea matematică informatică, intensiv informatică

Varianta 10

- Toate subiectele (I, II și III) 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. Indicați care dintre expresiile C/C++ de mai jos are valoarea 1 dacă și numai dacă numărul natural memorat în variabila întreagă n este divizibil cu 2 și cu 7. **(4p.)**
- a. $!(n \% 2 == 1) \ || \ (n \% 7 != 0)$ b. $(n \% 2 == 0) \ \&\& \ (n \% 7 != 0)$
c. $(n \% 2 == 0) \ || \ !(n \% 7 == 0)$ d. $(n \% 7 == 2) \ \&\& \ (n \% 2 == 7)$

2. Se consideră algoritmul alăturat descris în pseudocod:

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

- a) Scrieți numărul afișat dacă pentru variabila n se citește valoarea 64598. **(6p.)**
- b) Scrieți toate numerele de exact trei cifre care pot fi citite pentru variabila n astfel încât, pentru fiecare dintre acestea, numărul afișat în urma executării algoritmului să fie 24. **(4p.)**

```
citește n (număr natural nenul)
m ← 0
repetă
    c ← n % 10
    n ← [n / 10]
    dacă c > 5 atunci
        c ← [c / 2]
    m ← m * 10 + c
până când n = 0
scrie m
```

- c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască structura **repetă... până când** cu o structură repetitivă de alt tip. **(6p.)**
- d) Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

SUBIECTUL al II-lea

(30 de puncte)

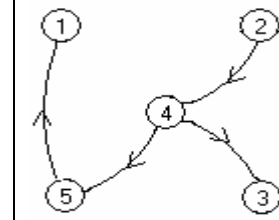
Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Se consideră declarațiile alăturate, în care variabila FIG memorează coordonatele, în planul xOy , ale centrului unui cerc, precum și lungimea razei acestuia. O expresie C/C++ care are valoarea 1 dacă și numai dacă centrul cercului se află pe prima bisectoare a sistemului de coordonate al planului xOy este: (4p.)

```
struct punct {  
    float x, y;  
};  
struct cerc {  
    struct punct centru;  
    float raza;  
} FIG;
```

- a. `FIG.centru(x)==FIG.centru(y)` b. `FIG.centru.x==FIG.centru.y`
c. `FIG.cerc.x==FIG.cerc.y` d. `FIG.punct.x==FIG.punct.y`

2. Pentru ca graful orientat cu 5 vârfuri, reprezentat în figura alăturată, să devină graf tare conex, numărul minim de arce care trebuie adăugate acestuia este: (4p.)



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

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

3. În secvența de instrucțiuni de mai jos, variabilele i și j sunt de tip întreg și se consideră că variabila A memorează elementele, inițial nule, ale unui tablou bidimensional, cu 5 linii și 5 coloane, numerotate de la 0 la 4.

Fără a utiliza și alte variabile decât cele menționate mai sus, scrieți o instrucțiune care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței, tabloul memorat în variabila A să aibă elementele din figura de mai jos.

```
for (i=0; i<5; i++)  
    for (j=0; j<5; j++)  
        .....
```

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

 (6p.)

4. Se consideră arborele cu rădăcină, având 10 noduri, numerotate de la 1 la 10, cu vectorul de „tați” $t=(3, 3, 8, 0, 8, 9, 9, 4, 4, 9)$. Enumerați toate nodurile arborelui care au exact doi fii. (6p.)

5. Se consideră un text cu cel mult 100 de caractere, format din cuvinte și numere, separate prin unul sau mai multe spații. Cuvintele sunt formate numai din litere ale alfabetului englez. Toate numerele sunt reale, pozitive și sunt formate numai din parte întreagă sau din parte întreagă și parte fracționară, separate prin punctul zecimal (.).

Scrieți un program C/C++ care citește de la tastatură textul, pe care îl transformă, eliminând din componența sa toate numerele care au parte fracționară. Programul afișează apoi pe ecran textul obținut.

Exemplu: dacă textul citit este

Reprezentat cu 4 zecimale PI este 3.1415 si e este 2.7182

se va afișa pe ecran textul:

Reprezentat cu 4 zecimale PI este si e este (10p.)

SUBIECTUL al III-lea

(30 de puncte)

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

1. Se consideră subprogramul `p`, definit alături. Valoarea lui `p(210, 49)` este:
- ```
int p(int m, int n)
{ if(n==0) return 1;
 return 1+p(n, m%n);
}
```
- (4p.)

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

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

2. Având la dispoziție cinci flori diferite, `lalea`, `narcisă`, `mac`, `frezie`, `garoafă`, se utilizează metoda backtracking pentru a obține toate posibilitățile de a forma un aranjament floral, știind că se folosesc toate cele cinci flori și contează ordinea de așezare a acestora. Primele patru soluții obținute sunt, în această ordine: (`lalea, narcisă, mac, frezie, garoafă`), (`lalea, narcisă, mac, garoafă, frezie`), (`lalea, narcisă, frezie, mac, garoafă`), (`lalea, narcisă, frezie, garoafă, mac`). Scrieți ultimele două soluții generate, în ordinea obținerii lor. (6p.)

3. Subprogramul `tablou` are trei parametri:
- `n` și `k`, prin care primește câte un număr natural ( $3 \leq n \leq 20$ ,  $2 \leq k \leq 10$ );
  - `a`, prin care furnizează un tablou unidimensional care memorează un șir crescător de `n·k` termeni, din mulțimea primelor `n` numere naturale nenule, fiecare astfel de număr fiind în șir de exact `k` ori.

Scrieți în limbajul C/C++ definiția completă a subprogramului.

**Exemplu:** dacă `n=4`, `k=3` atunci, după apel, `a=(1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4)`. (10p.)

4. Fișierul `BAC.TXT` conține, în ordine crescătoare, cel puțin două și cel mult 10000 de numere naturale. Numerele sunt separate prin câte un spațiu și au cel mult 9 cifre fiecare. Cel puțin un număr din fișier apare o singură dată.

a) Scrieți un program C/C++ care citește toate numerele din fișier și, printr-un algoritm eficient din punct de vedere al timpului de executare și al memoriei utilizate, determină și afișează pe ecran, în ordine strict crescătoare, separate prin câte un spațiu, numai numerele care apar o singură dată în fișier. (6p.)

**Exemplu:** dacă fișierul are conținutul de mai jos

1 1 2 2 2 7 10 10 10 10 21

pe ecran se afișează, în această ordine, numerele 7 21.

b) Descrieți în limbaj natural (3-4 rânduri) algoritmul utilizat la punctul a) și justificați eficiența acestuia. (4p.)