

Examenul de bacalaureat național 2015

Proba E. d)
Informatică
Limbajul C/C++

Varianta 9

Filiera teoretică, profilul real, specializările: matematică-informatică

matematică-informatică intensiv informatică

Filiera vocațională, profilul militar, specializarea matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv 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).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.

SUBIECTUL I

(30 de puncte)

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

1. Variabila întreagă **x** memorează un număr natural cu cel puțin patru cifre nenule distincte. Expresia C/C++ a cărei valoare este egală cu cifra sutelor acestui număr este: (4p.)

a. **x/100**

b. **x%100**

c. **(x/10)%10**

d. **(x/100)%10**

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.

S-a notat cu **a%b** restul împărțirii numărului natural **a** la numărul natural nenul **b** și cu **[c]** partea întreagă a numărului real **c**.

- a) Scrieți valoarea afișată dacă se citesc, în această ordine, numerele 7 și 2. (6p.)

- b) Dacă pentru variabila **k** se citește numărul 5, scrieți cea mai mică și cea mai mare valoare care pot fi citite pentru variabila **n** astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, valoarea afișată să fie 3. (4p.)

- c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind prima structură **cât timp...execută** cu o structură repetitivă de tip **pentru...execută**. (6p.)

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

```
citește n,k  
    (numere naturale, k>1)  
pm←0  
i←1  
cât timp i≤n execută  
    x←i  
    p←0  
    cât timp x%k=0 execută  
        x←[x/k]  
        p←p+1  
    ■  
    dacă p>pm atunci  
        pm←p  
    ■  
    i←i+1  
    ■  
scrie pm
```

SUBIECTUL al II-lea

(30 de puncte)

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

1. Variabila `c`, declarată alăturat, memorează titlul și prețul unei cărți. Expresia C/C++ a cărei valoare reprezintă prețul cărții respective majorat cu 50% este:

```
struct carte  
{ char titlu[21];  
  float pret;  
}c;
```

a. `c.pret*3/2` b. `pret.c*3/2` c. `c(pret)*3/2` d. `pret[c]*3/2`

2. Un arbore cu 37 de noduri, numerotate de la 1 la 37, are ca rădăcină nodul numerotat cu 1, iar tatăl fiecărui nod i ($i \in [2, 37]$) este numerotat cu partea întreagă a rădăcinii pătrate a lui i ($\lfloor \sqrt{i} \rfloor$). Numărul de frunze ale arborelui este:

a. 36 b. 31 c. 21 d. 6

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

3. Un graf neorientat cu 8 noduri, numerotate de la 1 la 8, are muchiile $[1,2]$, $[1,6]$, $[4,6]$, $[3,6]$, $[6,5]$, $[5,3]$, $[3,4]$, $[7,8]$, $[8,2]$. Enumerați trei noduri care nu aparțin niciunui ciclu în acest graf.

(6p.)

4. Fiind date două șiruri de caractere a și b , îl numim pe a **prefix** al lui b dacă a este egal cu b sau dacă b se poate obține din a prin alipirea la dreapta a unor noi caractere. Variabilele a și b pot memora câte un șir cu cel mult 20 de caractere. Știind că variabila b a fost inițializată cu un șir format dintr-un număr par de caractere, scrieți o secvență de instrucțiuni în urma executării căreia variabila a să memoreze un prefix al lui b a cărui lungime să fie jumătate din lungimea lui b .

Exemplu: dacă b memorează șirul aurari, atunci a memorează șirul aur.

(6p.)

5. Scrieți un program C/C++ care citește de la tastatură un număr natural, n ($n \in [2, 20]$), apoi n numere naturale din intervalul $[0, 10^4]$, reprezentând, de la stânga la dreapta, în această ordine, valorile elementelor aflate pe prima linie a unui tablou bidimensional cu n linii și n coloane. Programul construiește în memorie tabloul, inițializând celelalte elemente, astfel încât fiecare linie să se obțină prin permutarea circulară a elementelor liniei anterioare, de la stânga spre dreapta, cu o poziție, ca în exemplu.

Programul afișează pe ecran tabloul obținut, fiecare linie a tabloului pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.

Exemplu: dacă se citesc numerele $n=4$, apoi 1, 1, 3, 2, se obține tabloul alăturat.

1	1	3	2
2	1	1	3
3	2	1	1
1	3	2	1

(10p.)

SUBIECTUL al III-lea

(30 de puncte)

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

1. Utilizând metoda backtracking, se generează toate numerele naturale din intervalul $[100, 999]$ care au suma cifrelor egală cu 5. Primele cinci soluții obținute sunt, în această ordine 104, 113, 122, 131, 140. Utilizând același algoritm, se generează toate numerele naturale din intervalul $[1000, 9999]$ care au suma cifrelor egală cu 6. Al treilea număr generat este:

(4p.)

a. 1005

b. 1023

c. 1031

d. 1041

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

2. Subprogramul **F** este definit alăturat. Scrieți ce se afișează în urma apelului de mai jos.

F('d');

(6p.)

```
void F(char c)
{ if(c>='a')
  { cout<<c; | printf("%c",c);
    F(c-1);
  }
}
```

3. Șirul lui Fibonacci (1, 1, 2, 3, 5, 8, 13, 21,...) se definește astfel:

$f_1=1$, $f_2=1$ și $f_i=f_{i-1}+f_{i-2}$ pentru orice număr natural i , $i \geq 3$.

Subprogramul **Fibo** are un singur parametru, **n**, prin care primește un număr natural ($n \in [1, 30]$). Subprogramul returnează al **n**-lea termen impar al șirului lui Fibonacci.

Scrieți definiția completă a subprogramului.

Exemplu: dacă $n=6$, subprogramul returnează numărul 21.

(10p.)

4. Fișierul **bac.txt** conține un șir de cel mult un milion de numere naturale din intervalul $[0, 10^2]$, separate prin câte un spațiu.

Se cere să se determine toate perechile distincte formate din termeni ai șirului aflat în fișier, **x** și **y** ($y-x \geq 2$), astfel încât să nu existe niciun termen al șirului care să aparțină intervalului (x, y) . Numerele din fiecare pereche sunt afișate pe câte o linie a ecranului, în ordine strict crescătoare, separate printr-un spațiu, iar dacă nu există nicio astfel de pereche, se afișează pe ecran mesajul **nu exista**. Pentru determinarea numerelor cerute utilizați un algoritm eficient din punctul de vedere al timpului de executare.

Exemplu: dacă fișierul conține numerele

5 9 0 8 10 11 12 13 15 14 6 7 40 10 0 0 5 41 95 7

atunci pe ecran se afișează, nu neapărat în această ordine, perechile

0 5

15 40

41 95

a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia.

(4p.)

b) Scrieți programul C/C++ corespunzător algoritmului descris.

(6p.)