Examenul de bacalaureat naţional 2017
Proba E. d)
Informatică
Limbajul C/C++

Varianta 3

Filiera teoretică, profilul real, specializările: matematică-informatică
matematică-informatică intensiv informatică
Filiera vocaţională, 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 utilizaţi trebuie să respecte precizările din enunţ (bold), iar în lipsa unor precizări explicite, notaţiile trebuie să corespundă cu semnificaţiile 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, scrieţi pe foaia de examen litera corespunzătoare răspunsului corect.

1. Indicaţi expresia C/C++ care are valoarea **1** dacă şi numai dacă numărul natural memorat în variabila întreagă **x** are exact trei cifre. **(4p.)**

a. `x/1000==0 && x/100!=0`
b. `x/1000==0 || x/100!=0`
c. `x%1000==0 && x%100!=0`
d. `x%1000==0 || x%100!=0`

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.
S-a notat cu **a%b** restul împărţirii numărului natural **a** la numărul natural nenul **b**.

```
citeşte a,b
   (numere naturale, a≥2, b≥2)
dacă a>b atunci
| x←a
| a←b
| b←x
■
s←0
pentru x←a,b execută
| c←2
| cât timp x%c>0 execută
|| c←c+1
| ■
|s←s+c
■
scrie s
```

a) Scrieţi valoarea afişată dacă se citesc, în această ordine, numerele **7**, **4**. **(6p.)**

b) Scrieți patru seturi de date de intrare, numere naturale din intervalul **[2,9]**, care pot fi citite astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afişeze valoarea **7**. **(4p.)**

c) Scrieţi în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **cât timp...execută** cu o structură repetitivă cu test final. **(6p.)**

d) Scrieţi programul C/C++ corespunzător algoritmului dat. **(10p.)**

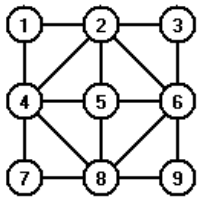
SUBIECTUL al II-lea **(30 de puncte)**

Pentru fiecare dintre itemii 1 şi 2 scrieţi pe foaia de examen litera corespunzătoare răspunsului corect.

1. Un arbore cu **9** noduri, numerotate de la **1** la **9**, este reprezentat prin vectorul de „taţi” (**5,4,6,0,3,2,6,9,7**). Rădăcina arborelui este: **(4p.)**

a. 1 **b.** 4 **c.** 6 **d.** 8

2. Într-un graf neorientat două cicluri sunt disjuncte dacă nu au niciun nod comun. Pentru graful neorientat cu **9** noduri, reprezentat alăturat, se construiește o mulțime formată din cicluri elementare, cu proprietatea că oricare două dintre acestea sunt disjuncte. Numărul maxim de cicluri dintr-o astfel de mulțime este: **(4p.)**



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

Scrieţi pe foaia de examen răspunsul pentru fiecare dintre cerinţele următoare.

3. Variabila **p** memorează simultan numărul de vârfuri ale unui poligon (număr natural din intervalul **[3,10²)**) și coordonatele vârfurilor acestuia (abscisa și ordonata) în sistemul de coordonate **xOy** (numere reale).
Știind că expresiile C/C++ de mai jos au ca valori numărul de vârfuri ale unui poligon, abscisa, respectiv ordonata primului său vârf, scrieți definiția unei structuri cu eticheta **poligon**, care permite memorarea datelor despre un poligon, și declarați corespunzător variabila **p**.

```
p.numar
p.varf[0].x
p.varf[0].y
```

(6p.)

4. Variabilele **i** şi **j** sunt de tip întreg, iar variabila **a** memorează un tablou bidimensional cu **9** linii şi **9** coloane, numerotate de la **0** la **8**, având iniţial toate elementele egale cu caracterul *.
Fără a utiliza alte variabile, scrieţi secvenţa de instrucţiuni de mai jos, înlocuind punctele de suspensie astfel încât, în urma executării secvenţei obţinute, variabila **a** să memoreze tabloul alăturat.

```
for(i=0;i<9;i++)
  for(j=0;j<9;j++)
    ...................
```

(6p.)

```
b a a a a a a a b
b b a a a a a b b
b b b a a a b b b
b b b b a b b b b
b b b b b b b b b
b b b b b b b b b
b b b b b b b b b
b b b b b b b b b
b b b b b b b b b
```

5. Se consideră un text cu cel mult **100** de caractere, în care cuvintele sunt formate din litere mari ale alfabetului englez și sunt separate prin câte un spațiu.
Scrieţi un program C/C++ care citeşte de la tastatură un text de tipul menţionat mai sus şi afișează pe ecran, pe câte un rând, cuvintele care cuprind cel puțin o notă muzicală. Dacă nu există astfel de cuvinte, se afișează pe ecran mesajul **nu exista**.
Notele muzicale sunt **DO**, **RE**, **MI**, **FA**, **SOL**, **LA**, **SI**.
Exemplu: pentru textul
REMI DOMINO SI KHANHOO SUNT DENUMIRI DE JOCURI CE AU ASPECTE SIMILARE
se afişează cuvintele de mai jos, nu neapărat în această ordine

```
REMI
DOMINO
SI
DENUMIRI
SIMILARE
```

(10p.)

SUBIECTUL al III-lea (30 de puncte)

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

1. Utilizând metoda backtracking se generează, în ordine crescătoare, toate numerele naturale pare cu trei cifre, cu proprietatea că nu există două cifre egale alăturate și suma cifrelor este **10**. Primele cinci numere generate sunt, în această ordine: **136**, **154**, **172**, **190**, **208**. Al şaselea număr generat este: **(4p.)**

a. **217** **b.** **226** **c.** **262** **d.** **280**

Scrieţi pe foaia de examen răspunsul pentru fiecare dintre cerinţele următoare.

2. Subprogramul **f** este definit alăturat. Scrieţi ce se afişează în urma apelului de mai jos.
f(4); **(6p.)**

```
void f(int n)
{ int i;
  if(n>0)
  { for(i=1;i<=n;i++)
      cout<<i;  |  printf("%d",i);
    f(n-1);
  }
}
```

3. Subprogramul **identice** are doi parametri, **a** și **b**, prin care primește câte un număr natural (**$10 \le a \le b \le 10^6$**). Subprogramul afișează pe ecran toate numerele naturale din intervalul **[a,b]** care au toate cifrele identice. Numerele afișate sunt separate prin câte un spațiu, iar dacă nu există astfel de numere, se afişează pe ecran mesajul **nu exista**.
Scrieţi definiţia completă a subprogramului.
Exemplu: pentru **a=700** și **b=1500** se afișează pe ecran
777 888 999 1111. **(10p.)**

4. Numim **inserare** a unui șir **A** într-un șir **B** introducerea, între două elemente ale șirului **B**, a tuturor elementelor lui **A**, pe poziții consecutive, în ordinea în care apar în **A**.
Fișierul **bac.in** conține numere naturale din intervalul **$[1,10^6]$**: pe prima linie numerele **m** și **n**, iar pe fiecare dintre următoarele două linii câte un șir de **m**, respectiv de **n** numere întregi ordonate **strict crescător**. Numerele aflate pe aceeași linie a fișierului sunt separate prin câte un spațiu, iar numerotarea elementelor în șiruri începe de la **1**.
Se cere să se afișeze pe ecran poziția din al doilea șir începând de la care poate fi inserat primul șir, astfel încât șirul obținut să fie strict crescător. Dacă nu există o astfel de poziție, se afișează pe ecran mesajul **imposibil**.
Proiectați un algoritm eficient din punctul de vedere al spațiului de memorie utilizat şi al timpului de executare.
Exemplu: dacă fișierul conține numerele
4 6
15 16 17 19
7 10 12 20 30 40
se poate obține șirul **7**, **10**, **12**, **15**, **16**, **17**, **19**, **20**, **30**, **40** și se afișează pe ecran **4**
iar dacă fișierul conține numerele
4 6
15 16 17 19
7 14 18 20 30 40
sau numerele
4 6
1 2 3 4
7 15 18 20 30 40
se afișează pe ecran mesajul **imposibil**
a) Descrieţi în limbaj natural algoritmul proiectat, justificând eficienţa acestuia. **(2p.)**
b) Scrieţi programul C/C++ corespunzător algoritmului descris. **(8p.)**