

Examenul de bacalaureat național 2014
Proba E. d)
Informatică
Limbajul C/C++

Simulare

Filiera teoretică, profilul real, specializarea științe ale naturii

- 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. Expresia **C/C++** alăturată are valoarea: (4p.) 7/2*5

- a. 0 b. 0.7 c. 15 d. 17.5

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.

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

a) Scrieți valoarea afișată dacă se citesc, în această ordine, numerele **21520** și **21523**. (6p.)

b) Dacă pentru **b** se citește valoarea **334**, scrieți patru valori naturale de trei cifre care pot fi citite pentru **a**, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze valoarea **3**. (4p.)

```
citește a,b  
    (numere naturale nenule, a<b)  
k←0  
nr←a  
cât timp nr≤b execută  
    | aux←nr  
    | cât timp aux>100 execută  
    | | aux←[aux/10]  
    | ■  
    | dacă nr%100=aux atunci  
    | | k←k+1  
    | | ■  
    | nr←nr+1  
    | ■  
scrie k
```

c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască prima structură **cât timp...execută** cu o structură repetitivă de tip **pentru...execută**. (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. O expresie C/C++ care are valoarea 1 este: **(4p.)**

- a. `ceil(7.81)==floor(8.17)-1` b. `ceil(7.81)==floor(8.17)`
c. `ceil(7.81)==ceil(8.71)` d. `floor(7.81)==floor(8.17)`

2. În secvențele de mai jos, notate cu **s1** și **s2**, toate variabilele sunt de tip întreg.

```
//s1
x=10; y=15;
while(x!=y)
    if(x>y) y=x+y;
    else x=x+y;
```

```
//s2
x=10; y=15;
while(y!=0) {
    r=x*y;
    x=y;
    y=r;
}
```

Variabila **x** memorează cel mai mare divizor comun al valorilor 10 și 15 în urma executării: **(4p.)**

- a. numai a secvenței **s1** b. numai a secvenței **s2**
c. atât a secvenței **s1**, cât și a secvenței **s2** d. niciuneia dintre cele două secvențe

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

3. Variabilele **luna** și **an** sunt de tip **int** și corespund unei date calendaristice: variabila **luna** are o valoare din intervalul **[1,12]**, iar variabila **an** are o valoare din intervalul **[1900,2200]**.

Scrieți o expresie C/C++ care să aibă valoarea 1, dacă data calendaristică menționată este cuprinsă în una dintre primele două luni ale anului **2014**, sau valoarea 0 în caz contrar. **(6p.)**

4. Se citește un număr natural nenul și se cere să se afișeze cel mai mic și cel mai mare divizor propriu pozitiv al său (diferit de 1 și de el însuși). Dacă numărul nu are cel puțin doi astfel de divizori, se afișează mesajul **Nu exista**.

Exemplu: pentru numărul 12 se afișează numerele 2 6, iar pentru numărul 9 se afișează mesajul **Nu exista**.

- a) Scrieți, în pseudocod, un algoritm de rezolvare pentru problema enunțată. **(10p.)**
b) Menționați rolul tuturor variabilelor care au intervenit în algoritmul realizat la punctul a) și indicați datele de intrare, respectiv datele de ieșire ale problemei enunțate. **(6p.)**

SUBIECTUL al III-lea

(30 de puncte)

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

1. În secvența de instrucțiuni de mai jos toate variabilele sunt de tip întreg.

```
for (i=0; i<9; i++)  
{ for (j=0; j<9; j++)  
    if (.....)  
        cout<<"1 "; | printf("1 ");  
    else  
        cout<<"2 "; | printf("2 ");  
    cout<<endl; | printf("\n");  
}
```

1	1	2	2	2	2	2	2	2
1	1	1	2	2	2	2	2	2
2	1	1	1	2	2	2	2	2
2	2	1	1	1	2	2	2	2
2	2	2	1	1	1	2	2	2
2	2	2	2	1	1	1	2	2
2	2	2	2	2	1	1	1	2
2	2	2	2	2	2	1	1	1
2	2	2	2	2	2	2	1	1

Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran valorile din figura de mai sus, în această ordine.
(4p.)

- a. `i==j && i==j-1 && i==j+1` b. `i==j && i==j-1 || i==j+1`
c. `i==j || j==i-1 && j==i+1` d. `i==j || j==i-1 || j==i+1`

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

2. Se consideră două tablouri unidimensionale: **A**, care are 3 elemente, și **B**, care are 4 elemente. Știind că unul dintre elementele tabloului **A** are valoarea 4 și unul dintre elementele tabloului **B** are valoarea 9, scrieți câte un exemplu de valori pentru elementele tabloului **A**, respectiv ale tabloului **B**, în ordinea în care ele pot apărea în fiecare tablou, astfel încât, prin metoda interclasării acestora, să se obțină tabloul unidimensional (4, 7, 9, 10, 14, 57, 80). **(6p.)**

3. Scrieți un program **C/C++** care citește de la tastatură un număr natural **n** ($2 < n < 50$) și cele **n** elemente ale unui tablou unidimensional, numere naturale cu cel mult nouă cifre. Cel puțin unul dintre elemente este nul.

Programul modifică tabloul în memorie, duplicând toate elementele nule ale acestuia, ca în exemplu, apoi afișează pe ecran, separate prin câte un spațiu, elementele tabloului obținut.

Exemplu: pentru **n=5** și tabloul (1, 0, 5, 0, 0, 7), programul afișează pe ecran:

1 0 0 5 0 0 0 0 7

(10p.)

4. Se consideră un șir ai cărui termeni sunt numere naturale nenule, de o singură cifră. Numim **număr asociat** al acestui șir un număr natural format cu termenii șirului, în ordinea în care aceștia apar în șir.

Exemplu: numărul asociat șirului 1, 2, 5, 3, 2 este 12532.

Fișierul text **bac.txt** conține un șir de cel puțin trei și cel mult 80 de termeni, numere naturale nenule, de o singură cifră, separate prin câte un spațiu.

Se cere determinarea unui șir obținut prin eliminarea unui singur termen din șirul aflat în fișier, astfel încât numărul asociat șirului obținut să fie maxim. Termenii șirului obținut se afișează pe ecran, separați prin câte un spațiu.

Se utilizează un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fișierul **bac.txt** conține șirul

9 8 5 6 2 3 4

atunci, pentru că numerele asociate șirurilor care se pot obține sunt 856234, 956234, 986234, 985234, 985634, 985624, 985623, pe ecran se afișează șirul:

9 8 6 2 3 4

a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**

b) Scrieți programul **C/C++** corespunzător algoritmului descris. **(6p.)**