

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

**Varianta 10**

**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. Variabila **x** este de tip întreg și poate memora un număr natural cu cel mult două cifre. Valoarea maximă pe care o poate avea expresia **C/C++** alăturată este: **(4p.)** **x%7**
- a. 6                      b. 14.14                      c. 93                      d. 693

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.

- a) Scrieți valoarea afișată dacă se citesc, în această ordine, numerele **10, 8, 11, 1, 21, 0**. **(6p.)**
- b) Scrieți un set de patru numere distincte din intervalul **[0, 9]** care pot fi citite astfel încât, în urma executării algoritmului, să se afișeze valoarea **0**. **(4p.)**
- c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască prima structură **repetă...până când** cu o structură repetitivă cu test inițial. **(6p.)**

```
n ← 0
repetă
    citește x
    (număr natural)
    a ← 0
    b ← 1
    repetă
        c ← a + b
        a ← b
        b ← c
    până când c ≥ x
    dacă x = c atunci
        n ← n + 1
    ■
până când x = 0
scrie n
```

- 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. Indicați cel mai mare număr cu două cifre pe care îl poate memora variabila întreagă  $x$  astfel încât expresia  $C/C++$  alăturată să aibă valoarea 1. **(4p.)**

$\text{sqrt}(x/10+x\%10)==4$

- a. 16                      b. 17                      c. 88                      d. 97

2. Se consideră secvențele de mai jos, notate cu  $s1$ ,  $s2$  și  $s3$ , în care toate variabilele sunt întregi, iar variabilele  $k$  și  $n$  memorează câte un număr natural cu cel mult două cifre ( $k < n$ ).

//s1  
 $p = (n * n - k * k + n + k) / 2 \% 10;$

//s2  
 $p = 0;$   
 $\text{for}(i = k; i \leq n; i++)$   
 $p = (p + i) \% 10;$

//s3  
 $p = k;$   
 $\text{for}(i = k + 1; i \leq n; i++)$   
 $p = p \% 10 + i;$

Variabila  $p$  memorează ultima cifră a sumei numerelor naturale distincte din intervalul  $[k, n]$  în urma executării, independent, numai a secvențelor: **(4p.)**

- a.  $s1$  și  $s2$                       b.  $s2$  și  $s3$                       c.  $s2$                       d.  $s1$

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

3. Variabila  $s$  este de tip `char` și memorează o literă mică a alfabetului englez. Scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran litera care îi urmează imediat în alfabetul englez celei memorate în variabila  $s$ , dacă aceasta este o vocală din mulțimea  $\{a, e, i\}$ , sau litera care o precede imediat în alfabetul englez pe aceasta în caz contrar.

**Exemplu:** dacă  $s$  memorează litera  $e$  se afișează  $f$ , iar dacă  $s$  memorează litera  $c$  se afișează  $b$ . **(6p.)**

4. Se citește un număr natural,  $n$  ( $n \geq 2$ ), și se cere să se afișeze toate tripletele de numere naturale  $(x, y, z)$  cu proprietatea că  $x < y < z$  și  $x \cdot y + y \cdot z = n$ . Numerele din fiecare triplet se afișează separate prin câte o virgulă și încadrate între paranteze rotunde, ca în exemplu.

**Exemplu:** pentru  $n=8$  se afișează, nu neapărat în această ordine, tripletele:

$(0, 1, 8)$   $(0, 2, 4)$   $(1, 2, 3)$

a) Scrieți, în pseudocod, algoritmul 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. Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea  $x=21$ , se aplică metoda căutării binare, iar succesiunea de elemente ale tabloului a căror valoare se compară cu valoarea lui  $x$  pe parcursul aplicării metodei indicate este: **49, 16, 21**. Elementele tabloului pot fi (în ordinea în care apar în tablou): **(4p.)**
- a. **(16, 17, 21, 29, 49, 80, 95)**                      b. **(4, 16, 21, 49, 56, 70, 85)**
- c. **(7, 9, 10, 16, 21, 45, 49)**                      d. **(16, 20, 21, 49, 50, 56, 59)**

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

2. În secvența alăturată toate variabilele sunt de tip întreg, iar numerele citite sunt naturale. Scrieți secvența înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei **ok** să fie **1** dacă toate valorile citite au fost strict mai mici decât **2014**, sau **0** altfel. **(6p.)**
- ```
ok=.....;
for (i=1; i<=10; i++)
{ cin>>x; | scanf("%d", &x);
  .....
}
```
3. Scrieți un program **C/C++** care citește de la tastatură un număr natural,  $n$  ( $2 \leq n \leq 50$ ), și construiește în memorie un tablou unidimensional cu  $n$  elemente, astfel încât, parcurgându-l de la stânga la dreapta, se obține șirul primelor  $n$  numere naturale, pare, care **NU** sunt divizibile cu **5**, ordonat strict crescător, ca în exemplu. Programul afișează pe ecran elementele tabloului obținut, separate prin câte un spațiu. **Exemplu:** dacă  $n=7$ , se obține tabloul **(2, 4, 6, 8, 12, 14, 16)**. **(10p.)**
4. Fișierul **bac.txt** conține pe prima linie un număr natural,  $n$  ( $1 \leq n \leq 10^6$ ), iar pe a doua linie cel mult **1000000** de numere naturale de forma  $10^p$  ( $0 \leq p \leq 9$ ), separate prin câte un spațiu. Se cere să se afișeze pe ecran numărul care ar apărea pe poziția  $n$  în șirul ordonat strict crescător obținut din toate numerele **distincte** aflate pe a doua linie a fișierului. Dacă șirul are mai puțin de  $n$  termeni distincți, se afișează pe ecran mesajul **Nu exista**. Pentru determinarea numărului cerut se utilizează un algoritm eficient din punctul de vedere al timpului de executare. **Exemplu:** dacă fișierul **bac.txt** conține numerele
- ```
4
100 100000 1 100000 1000 100 10 100
```
- atunci pe ecran se afișează valoarea **1000**
- a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**
- b) Scrieți programul **C/C++** corespunzător algoritmului descris. **(6p.)**