

Examenul național de bacalaureat 2021  
Proba E. d)  
INFORMATICĂ  
Limbajul C/C++

Varianta 1

Filieră teoretică, profil real, specializare științe ale naturii

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei ore.
- Identificatorii utilizați în rezolvări 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). Datele de intrare se consideră corecte, validarea lor nefiind necesară.

**SUBIECTUL I** **(20 de puncte)**

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

- Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă valorile variabilelor întregi  $x$  și  $y$  sunt numere pare.
  - $x\%2==0 \ \&\& \ (y+1)\%2!=0$
  - $(x-y)/2==0$
  - $(x+y)\%2==0 \ \&\& \ (x-y)\%2==0$
  - $x\%2==y\%2$
- Tablourile unidimensionale  $A$  și  $B$  au valorile  $A=(1,7,10,18,32)$  și  $B=(2,5,12,16,49)$  și se interclasează în ordine crescătoare, fiind parcurse de la stânga la dreapta. Pentru a determina al 4-lea element obținut în urma interclasării, se compară elementul cu valoarea  $xa$  din  $A$  cu elementul cu valoarea  $xb$  din  $B$ . Indicați valorile lui  $xa$  și  $xb$ .
  - $xa=7$  și  $xb=5$
  - $xa=7$  și  $xb=12$
  - $xa=10$  și  $xb=16$
  - $xa=18$  și  $xb=16$
- Variabilele întregi  $fa$  și  $fb$  memorează numărătorul, respectiv numitorul unei fracții. Indicați o secvență de instrucțiuni C/C++ care să memoreze în variabilele întregi  $sa$  și  $sb$  numărătorul, respectiv numitorul fracției obținute prin scăderea fracției menționate mai sus din fracția  $\frac{2020}{2021}$ .
  - $sa=2020*fb-2021*fa;$   
 $sb=fb*2021;$
  - $sa=2020*(fa-fb);$   
 $sb=fb*2021;$
  - $sa=2020-fa;$   
 $sb=fb*2021;$
  - $sa=2020/fb-2021/fa;$   
 $sb=fb/2021;$
- Variabilele  $x$  și  $y$  sunt de tip real,  $x$  are valoarea 3.5, iar  $y$  are valoarea 7.2. Indicați expresia C/C++ a cărei valoare este 3.
  - `floor(-x-y)`
  - `floor(x+y)`
  - `floor(x-y)`
  - `floor(y-x)`
- Șirul lui Fibonacci are termenii 1, 1, 2, 3, 5, 8 .... În secvența C/C++ alăturată toate variabilele sunt de tip întreg. Indicați expresia care poate înlocui punctele de suspensie, astfel încât, în urma executării secvenței obținute, variabila  $z$  să aibă valoarea 1 dacă numerele nenule memorate inițial în variabilele  $x$  și  $y$  ( $x < y$ ) sunt termeni consecutivi ai șirului lui Fibonacci, sau valoarea 0 altfel.

```
while(x>0)
{
    z=y-x;
    y=x;
    x=z;
}
if(.....) z=1;
else z=0;
```

  - $x==0 \ || \ y==1$
  - $x==1 \ || \ y==0$
  - $x==0 \ \&\& \ y==1$
  - $x==1 \ \&\& \ y==1$

**SUBIECTUL al II-lea**

**(40 de puncte)**

1. Algoritmul alăturat este reprezentat în pseudocod.  
S-a notat cu  $a \leftrightarrow b$  operația de interschimbare a valorilor variabilelor  $a$  și  $b$ .
- a. Scrieți ce se afișează în urma executării algoritmului dacă se citesc, în această ordine, numerele 8 și 5. (6p.)
- b. Dacă pentru variabila  $x$  se citește valoarea 10, scrieți două numere care pot fi citite pentru variabila  $y$ , astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, cifra 2 să fie afișată doar de trei ori. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat, fără a utiliza eventuale subprograme predefinite pentru interschimbare. (10p.)
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind structura **pentru...execută** cu o structură repetitivă cu test inițial. (6p.)
2. Elementele unui tablou unidimensional sunt, în această ordine, (2, 7, 10, 12, 16, 36, 45). Pentru a verifica dacă în tablou există elementul cu valoarea  $x=8$ , se aplică metoda căutării binare. Scrieți succesiunea de elemente a căror valoare se compară cu valoarea lui  $x$  pe parcursul aplicării metodei indicate. (6p.)
3. În secvența de mai jos toate variabilele sunt de tip întreg, iar de la tastatură se citesc 10 numere naturale.  
Scrieți secvența înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei  $maxp$  să fie egală cu cel mai mare dintre numerele citite care sunt pare și au două cifre, sau să fie egală cu -1 dacă nu există astfel de numere. (6p.)

```

citește x,y
    (numere naturale nenule)
┌dacă x>y atunci x↔y
└─┘
nr←1
┌pentru i←y,x,-1 execută
│  scrie 1
│ ┌dacă nr≥x atunci
│ │ scrie 2
│ └─┘
│ nr←nr*3
│ scrie 1
└─┘

```

```

maxp=.....;
for(i=1;i<=10;i++)
{ cin>>x; | scanf("%d",&x);
  .....
}

```

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Se citește un număr natural  $n$  ( $n \geq 2$ ) și se cere să se scrie suma divizorilor primi ai lui  $n$  care apar la o putere impară în descompunerea în factori primi a acestuia.  
Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.  
**Exemple:** pentru  $n=360$ , se scrie 7 ( $360=2^3 \cdot 3^2 \cdot 5^1$ , deci suma este 2+5), iar pentru  $n=16$ , se scrie 0. (10p.)
2. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul  $[1, 10^2]$ ,  $n$  și  $k$ , și cele  $n$  elemente ale unui tablou unidimensional, numere naturale din intervalul  $[0, 10^9]$ . Programul afișează pe ecran, separate prin câte un spațiu, primele  $k$  elemente ale tabloului care au cifra unităților 0, sau doar mesajul **nu exista** dacă nu există  $k$  astfel de elemente.  
**Exemplu:** pentru  $n=8$ , tabloul (23, 70, 61, 8, 0, 50, 742, 10) și  $k=3$ , se afișează pe ecran numerele 70 0 50 (10p.)
3. Numărul natural  $a$  se numește **sufix** al numărului natural  $b$  dacă  $a$  este egal cu  $b$  sau dacă  $b$  se poate obține din  $a$  prin alipirea la stânga a unor noi cifre.  
Fișierul **bac.txt** conține pe prima linie un număr natural  $x$  ( $x \in [100, 999]$ ), iar pe a doua linie un șir de cel mult  $10^5$  numere naturale din intervalul  $[0, 10^9]$ . Numerele din șir sunt separate prin câte un spațiu.  
Se cere să se afișeze pe ecran ultimul termen al șirului care îl are drept sufix pe numărul  $x$ . Dacă nu există un astfel de termen, se afișează pe ecran mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.  
**Exemplu:** dacă fișierul conține  
210  
3445 210 893210 1245 1210 3210 15210 67120 20210 12  
ecran se afișează 20210
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)  
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)