

Examenul de bacalaureat național 2020
Proba E. d)
Informatică
Limbajul C/C++

Testul 17

Filieră teoretică, profil real, specializare științe ale naturii

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 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.

1. Variabila x este de tip întreg și memorează un număr nenul. Indicați expresia C/C++ cu valoarea 1 pentru orice multiplu al lui 2020 memorat în variabila x .
- a. $x / (x / 2020) == 0$ b. $x / (x \% 2020) == 0$ c. $x \% (x \% 2020) == 0$ d. $x \% (x / 2020) == 0$
2. Variabilele i și j sunt de tip întreg. Indicați expresia care poate înlocui zona punctată astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran valorile alăturate.
- ```
for (i=0; i<4; i++)
{ for (j=0; j<5; j++)
 cout<<.....<<" "; | printf("%d ",);
 cout<<endl; | printf("\n");
}
```
- 20 16 12 8 4  
19 15 11 7 3  
18 14 10 6 2  
17 13 9 5 1
- a.  $20 - 4 * i - j$       b.  $20 - 4 * i + j$       c.  $20 - i - 4 * j$       d.  $20 - i + 4 * j$
3. Tabloul unidimensional  $A$  are elementele:  $A = (2, 20, 27, 36, 50)$ , iar în urma interclasării lui descrescătoare cu tabloul unidimensional  $B$  se obține tabloul cu elementele  $(63, 50, 45, 36, 27, 20, 8, 5, 3, 2)$ . Indicați elementele tabloului  $B$ , în ordinea apariției lor în acesta.
- a.  $(1, -15, -19, 9, 13)$       b.  $(51, 25, -19, -31, -47)$   
c.  $(63, 45, 8, 5, 3)$       d.  $(63, 45, 8, 5, 3, 50, 36, 27, 20, 2)$
4. Indicați expresia a cărei valoare este 20.
- a. `floor(21.19)`      b. `floor(20.19)`      c. `floor(19.20)`      d. `floor(21.20)`
5. În secvența de instrucțiuni alăturată, toate variabilele sunt întregi. Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila  $x$  să aibă o valoare egală cu cel mai mare divizor comun al numerelor 50 și 2020.
- ```
x=2020; y=50;  
while (x!=y)  
{ if(...) x=x-y;  
  else y=y-x;  
}
```
- a. $x - y > 0$ b. $x + y > 0$ c. $x / y == 0$ d. $x \% y == 0$

SUBIECTUL al II-lea (40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.
- S-a notat cu $a \% b$ restul împărțirii numărului natural a la numărul natural nenul b .
- a. Scrieți ce se afișează în urma executării algoritmului dacă se citesc, în această ordine, numerele 3 și 12. (6p.)
- b. Scrieți două seturi de date din intervalul $[1, 10]$ care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze numărul 20. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat structura `cât timp...execută` cu o structură repetitivă de alt tip. (6p.)
- ```
citește x,y
(numere naturale $x \leq y$)
 $i \leftarrow x$; $j \leftarrow y$; $s \leftarrow 0$
cât timp $i \leq j$ execută
┌─dacă $i \% 2 = 0$ atunci
│ $s \leftarrow s + j$
└─
┌─dacă $j \% 2 = 0$ atunci
│ $s \leftarrow s + i$
└─
 $i \leftarrow i + 1$; $j \leftarrow j - 1$
scrie s
```

2. Pentru un obiectiv turistic din evidența unei agenții se memorează o perioadă din an în care se recomandă să fie vizitat, formată din zile consecutive, între două date (ziua și luna de început, respectiv ziua și luna finală). Variabilele întregi `zi1` și `luna1` memorează ziua, respectiv luna corespunzătoare uneia dintre cele două date precizate, iar variabilele întregi `zi2` și `luna2` memorează ziua, respectiv luna corespunzătoare celeilalte date precizate. Scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran, pe rânduri separate, cele două date, în ordine cronologică; pentru fiecare dată se afișează ziua, apoi luna, separate printr-un spațiu. **(6p.)**
3. Variabila `a` este de tip `char`, iar variabilele `i` și `ok` sunt de tip întreg. De la tastatură se citesc numai litere mici ale alfabetului englez.

Fără a utiliza alte variabile, scrieți secvența înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila `ok` să aibă valoarea 1 dacă toate literele citite sunt vocale, sau valoarea 0 altfel. Se consideră vocale literele `a`, `e`, `i`, `o`, `u`.

```
.....
for(i=1;i<=11;i++)
{ cin>>a; | scanf("%c",&a);
.....
}
```

**Exemplu:** dacă se citesc literele `o a i a a i a e a e i` variabila `ok` va memora 1.

**(6p.)**

### SUBIECTUL al III-lea

**(30 de puncte)**

1. Se citește un număr natural, `n`, și se cere să se scrie cea mai mare cifră impară din scrierea acestuia, sau `-1` dacă nu există astfel de cifre. Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.

**Exemplu:** dacă `n=5672883`, se scrie 7.

**(10p.)**

2. Scrieți un program C/C++ care citește de la tastatură numere naturale: `n` ( $n \in [2, 20]$ ) și cele `n` elemente ale unui tablou unidimensional, valori din intervalul  $[10^2, 10^3]$ . Programul transformă tabloul în memorie, duplicând fiecare număr format numai din cifre impare, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, separate prin câte un spațiu, iar dacă nu există numere duplicate, se afișează pe ecran mesajul `nu exista`.

**Exemplu:** pentru `n=6` și tabloul `(100, 315, 507, 160, 375, 843)` se obține tabloul `(100, 315, 315, 507, 160, 375, 375, 843)`.

**(10p.)**

3. Fișierul text `bac.txt` conține numere naturale din intervalul  $[1, 10^4]$ : pe prima linie un număr `n`, pe a doua linie un șir de `n` numere, iar pe fiecare dintre următoarele linii, până la finalul fișierului, câte un termen al unui al doilea șir. Numerele aflate pe aceeași linie a fișierului sunt în ordine crescătoare și sunt separate prin câte un spațiu.

Se cere să se afișeze pe ecran numărul de termeni ai celui de al doilea șir care **NU** aparțin primului șir. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

**Exemplu:** dacă fișierul conține numerele alăturate, se afișează pe ecran 3.

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**

b. Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**

|    |             |
|----|-------------|
| 5  | 5           |
| 4  | 4 8 9 16 25 |
| 3  | 3           |
| 8  | 8           |
| 16 | 16          |
| 7  | 7           |
| 25 | 25          |
| 10 | 10          |