

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

Testul 1

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.

- Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul natural memorat în variabila întregă n este divizibil cu 2 și cu 5.
 - $!(n\%2==1 \ || \ n\%5!=0)$
 - $!(n/2==1 \ \&\& \ n/5!=0)$
 - $n\%2==0 \ || \ !(n\%5==0)$
 - $n/2==0 \ \&\& \ !(n/5==0)$
- Pentru a verifica dacă în tabloul unidimensional (4, 7, 9, 15, 16, 20, 27) există elementul cu valoarea $x=18$ se aplică metoda căutării binare. Succesiunea de elemente a căror valoare se compară cu x pe parcursul aplicării metodei este:
 - 4, 16, 20
 - 15, 20, 16
 - 15, 16, 20
 - 4, 9, 16, 27
- Tablourile unidimensionale **A** și **B** au valorile: **A**=(20, 16, 10, 7, 2) și **B**=(56, 15, 14, 11, 1). În urma interclasării lor în ordine descrescătoare se obține tabloul cu elementele:
 - (20, 15, 10, 7, 1)
 - (56, 20, 16, 15, 14, 10, 11, 7, 2, 1)
 - (56, 15, 14, 11, 1, 20, 16, 10, 7, 2)
 - (56, 20, 16, 15, 14, 11, 10, 7, 2, 1)
- O expresie C/C++ care are valoarea 1 este:
 - $ceil(20.20) - 1 == floor(19.20)$
 - $ceil(20.19) == floor(20.20)$
 - $ceil(20.20) == ceil(19.20)$
 - $floor(20.19) == floor(20.20)$
- În secvența de instrucțiuni alăturată, toate variabilele sunt de tip întreg. O expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței, variabila m să aibă o valoare egală cu cel mai mare divizor comun al numerelor 2020 și 1020 este:

	<pre>m=2020; n=1020; while (...) { x=m%n; m=n; n=x; }</pre>		
a. $m!=0$	b. $m\%n!=0$	c. $n!=0$	d. $n!=m$

SUBIECTUL al II-lea (40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.**
S-a notat cu $a\%b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .
 - Scrieți valoarea care se afișează în urma executării algoritmului dacă se citesc, în această ordine, numerele 12345, 780, 921, 4013, 75, 100214. (6p.)
 - Dacă pentru n se citește numărul 49, scrieți două seturi de date care pot fi citite în continuare astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze 49. (6p.)
 - Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
 - Scrieți în pseudocod un algoritm, echivalent cu cel dat, care să conțină o singură instrucțiune repetitivă. (6p.)

	<pre>citește n (număr natural) p←1; m←0; k←0 cât timp n≠0 execută citește x (număr natural) pentru i←1, k execută x←[x/10] ■ dacă x≠0 atunci c←x%10 altfel c←n%10 ■ m←c*p+m n←[n/10] p←p*10; k←k+1 ■ scrie m</pre>
--	--

2. Variabilele următoare memorează coordonatele reale (abscisa și ordonata), în planul xOy , ale vârfurilor unui triunghi: tAx și tAy abscisa și respectiv ordonata vârfului A , tBx și tBy abscisa și respectiv ordonata vârfului B , iar tCx și tCy abscisa și respectiv ordonata vârfului C . Declarați corespunzător variabilele și scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran pătratul lungimii laturii AB a triunghiului. (6p.)
3. În secvența alăturată, variabila a memorează un caracter, iar variabilele i și k sunt de tip întreg. Scrieți valoarea variabilei k în urma executării secvenței știind că se citește de la tastatură, în această ordine, literele **V I C T O R I E** (6p.)

```
k=0;
for(i=1;i<=8;i++)
{ cin>>a; | scanf("%c",&a);
  if(a>='A' && a<='Z') k=k+1;
  else k=k-1;
}
```

SUBIECTUL al III-lea

(30 de puncte)

1. Se citește un număr natural nenul n și se cere să se scrie, în această ordine, numerele d și p , reprezentând divizorul prim, d , care apare la cea mai mare putere, p , în descompunerea în factori primi a lui n ; dacă există mai mulți astfel de divizori se afișează cel mai mare dintre ei. Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate. **Exemplu:** dacă $n=10780$, atunci se scriu numerele 7 2 ($10780=2^2 \cdot 5 \cdot 7^2 \cdot 11$). (10p.)
2. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul $[2, 20]$, n și k , și construiește în memorie un tablou unidimensional cu $n \cdot k$ elemente astfel: tabloul memorează un șir crescător de termeni cu proprietatea că primul termen este k , fiecare valoare apare în șir de exact k ori și oricare doi termeni alăturați au valori egale sau consecutive. Programul afișează pe ecran tabloul construit, cu valorile separate prin câte un spațiu. **Exemplu:** dacă $n=4$ și $k=3$, se afișează pe ecran tabloul alăturat. (10p.)
- | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 3 | 3 | 4 | 4 | 4 | 5 | 5 | 5 | 6 | 6 | 6 |
|---|---|---|---|---|---|---|---|---|---|---|---|
3. Se consideră șirul $1, 1, 2, 5, 14, 41, 122, 365, 1094 \dots$ definit astfel: $f_1=f_2=1$, $f_n=3 \cdot f_{n-1}-1$ (unde n este un număr natural $n \geq 3$): Se citește de la tastatură un număr natural x ($x \leq 10^9$), valoare a unui termen din șirul dat, și se cere să se scrie în fișierul text **bac.txt**, în ordine descrescătoare, separați prin câte un spațiu, toți termenii șirului care sunt mai mici sau egali cu x . Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare. **Exemplu:** dacă se citește numărul 365 fișierul **bac.txt** conține numerele 365 122 41 14 5 2 1 1
- a. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)
- b. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)