

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

Testul 8

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 memorat în variabila întregă x aparține reuniunii de intervale $[-2020, -2019] \cup [2019, 2020]$.
 - $x >= -2020 \ \&\& \ x <= -2019 \ \&\& \ x >= 2019 \ \&\& \ x <= 2020$
 - $!(x < -2020 \ || \ x > -2019) \ || \ !(x < 2019 \ || \ x > 2020)$
 - $x >= -2020 \ || \ x <= -2019 \ || \ x >= 2019 \ || \ x <= 2020$
 - $!(x < -2020 \ \&\& \ x > 2020 \ \&\& \ x > -2019 \ || \ x < 2019)$
- Pentru a verifica dacă în tabloul unidimensional $(70, 64, 24, 14, 12, 8, 2)$ există elementul cu valoarea x se aplică metoda căutării binare. Știind că valoarea x a fost comparată cu un singur element al tabloului pe parcursul aplicării metodei, x ar putea fi:
 - 70
 - 14
 - 64 sau 2
 - 8 sau 2
- Tablourile unidimensionale A și B au elementele: $A = (1, 3, 6, 17, 18)$, iar $B = (2, 9, 10, 21, 49)$. În urma interclasării lor în ordine crescătoare se obține tabloul cu elementele:
 - $(1, 2, 3, 6, 9, 10, 17, 18, 21, 49)$
 - $(1, 2, 3, 9, 6, 10, 17, 21, 18, 49)$
 - $(1, 3, 6, 17, 18, 2, 9, 10, 21, 49)$
 - $(3, 12, 16, 38, 67)$
- Indicați o expresie C/C++ care are valoarea 1.
 - $\text{floor}(20.02 * 100) == \text{ceil}(20.20) * 100$
 - $\text{floor}(20.02) == \text{ceil}(2.020) * 10$
 - $\text{floor}(20.02) == \text{floor}(2.020) * 10$
 - $\text{floor}(20.20) / 10 == \text{ceil}(2.002)$
- Variabilele x , y , z , w și r sunt de tip întreg, iar r memorează inițial valoarea 0. Indicați o secvență echivalentă cu cea de mai jos.
`if (x==y){ r=1; if(z==w) r=2; } else r=3;`
 - `if (x==y || z!=w) r=1; else if(x!=y || z==w) r=2; else if(x!=y) r=3;`
 - `if (x==y || z!=w) r=1; else if(x==y || z==w) r=2; else r=3;`
 - `if (x==y && z!=w) r=1; else if(x!=y && z==w) r=2; else if(x!=y) r=3;`
 - `if (x==y && z!=w) r=1; else if(x==y && z==w) r=2; else r=3;`

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 și cu $[c]$ partea întreagă a numărului real c .
- a. Scrieți valoarea afișată dacă se citește, în această ordine, numerele 201920 și 20. (6p.)
- b. Dacă numărul citit pentru y este 0, scrieți două numere din intervalul $[10^2, 10^3)$ care pot fi citite pentru x , astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, numărul afișat să fie 9. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind structura **repetă... până când** cu o structură repetitivă cu test inițial. (6p.)
- ```
citește x, y
(numere naturale)
repetă
 c ← x % 10
 x ← [x / 10]
 dacă c ≠ 0 atunci
 dacă y % 10 < c atunci
 y ← y * 10 + c
 altfel
 y ← y * 10 - c
 până când x = 0
 scrie y
```
2. Despre un elev se memorează în variabila  $nrm$  numărul matricol (număr natural cu cel mult 3 cifre), iar în variabilele  $nota1$  și  $nota2$  câte o notă, număr natural din intervalul  $[1, 10]$ . Scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe ecran media celor două note, cu două zecimale exacte, urmate, pe rândul următor, de numărul matricol al elevului menționat. (6p.)
3. Variabila  $i$  este de tip întreg, iar celelalte variabile sunt de tip  $char$ . Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)
- ```
c1='s'; c2='o'; c3='u';  
cout << c2 << c3 << endl; | printf("%c%c\n", c2, c3);  
for(i=0; i<4; i++)  
{ if(i%2==0)  
  { c=c1+i-1;  
    if(i==0) c1=c;  
  }  
  else c=c1+3*(2*i/3-1);  
  cout << c; | printf("%c", c);  
}
```

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 . Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate. **Exemplu:** pentru $n=12$ se scrie 5 (divizorii primi ai lui 12 sunt 2 și 3). (10p.)
2. Scrieți un program C/C++ care citește de la tastatură numere naturale: n ($n \in [2, 20]$), k ($k \in [1, n]$) și cele n elemente ale unui tablou unidimensional, numere din intervalul $[0, 10^9]$. Programul transformă tabloul în memorie, deplasând circular spre stânga, cu câte o poziție, primele k elemente, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, separate prin câte un spațiu. **Exemplu:** pentru $n=7$, $k=4$ și tabloul (1, 3, 5, 7, 9, 5, 8) se obține tabloul (3, 5, 7, 1, 9, 5, 8). (10p.)
3. Fișierul `bac.in` conține un șir de cel mult 10^6 numere naturale din intervalul $[0, 10^9]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran cifrele cu proprietatea că în șir se află un număr maxim de termeni care au cifra unităților egală cu acestea. Cifrele afișate sunt separate prin câte un spațiu. Proiectați un algoritm eficient din punctul de vedere al timpului de executare. **Exemplu:** dacă fișierul `bac.in` conține numerele 112 12 5 25 88 15 2 19 32 179 35 621 pe ecran se afișează cifrele de mai jos (sunt câte 4 termeni care au cifra unităților 2, respectiv 5):
2 5
a. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)
b. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)