

Examenul național de bacalaureat 2023
Proba E. d)
Informatică
Limbajul C/C++

Varianta 5

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 expresia C/C++ care are cea mai mare valoare, comparativ cu celelalte trei expresii.
a. $20 * 23 / (2 * 2)$ b. $20 / 2 * 23 / 2$ c. $(20 * 23) / 2$ d. $(20 * 23) / 2 * 2$
- În secvența alăturată toate variabilele sunt întregi. Indicați expresia care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze:
11001
a. $n * 2$ b. $n / 2$ c. $n + 2$ d. $n - 2$

```
n=19;
while (n!=0)
{ cout<<n%2; | printf("%d",n%2);
  n=.....;
}
```
- Aplicând metoda căutării binare pentru a verifica dacă în tabloul unidimensional (2011, 2013, 2015, 2017, 2019, 2021, 2023) există elementul cu valoarea x , aceasta a fost comparată cu trei elemente. Indicați două valori posibile ale lui x .
a. 2019, 2025 b. 2017, 2019 c. 2013, 2017 d. 2011, 2013
- În secvențele de mai jos, toate variabilele sunt de tip întreg. Indicați secvența de instrucțiuni care interschimbă valorile memorate în variabilele întregi x și y .
a. $x=x-y;$
 $y=x+y;$
 $x=x+y;$ b. $x=x-y;$
 $y=x-y;$
 $x=x+y;$ c. $y=x+y;$
 $x=y-x;$
 $y=y-x;$ d. $y=x+y;$
 $x=x+y;$
 $y=y-x;$
- Indicați un număr care poate fi memorat în variabila reală x , astfel încât expresia C/C++ alăturată să aibă valoarea 1.
a. 0.4 b. 1.16 c. 1.4 d. 1.84

```
ceil(x) - x < 0.2
```

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 întregă a numărului real c .

- Scrieți valoarea afișată dacă se citește numărul 6907512. (6p.)
- Scrieți două valori distincte din intervalul [100, 999] care pot fi citite astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze o valoare identică cu cea citită. (6p.)
- Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind structura **cât timp...execută** cu o structură repetitivă cu test final. (6p.)

```
citește x (număr natural)
p ← 1; m ← -1
cât timp p ≤ x execută
  c ← [x/p] % 10
  dacă c > m atunci
    m ← c; p ← p * 10
  altfel
    x ← [x / (p * 10)] * p + x % p
dacă m ≥ 0 atunci scrie x
altfel scrie "nul"
```

- Scrieți un exemplu de valori distincte pentru x , y și z , astfel încât, în urma interclasării în ordine crescătoare a tablourilor $A = (2019, z, x, 29, 17)$ și $B = (2000, 45, y, 32, 4)$, care nu au elemente comune, valorile x , y și z să ocupe poziții consecutive, în această ordine, în tabloul rezultat. (6p.)

3. O florărie achiziționează două soiuri de lalele. Pentru fiecare soi se memorează datele: **cod1** (o literă mare a alfabetului englez) și prețul unui fir, în lei (număr natural). Variabilele **cod1** și **pret1** memorează datele primului soi, iar variabilele **cod2** și **pret2** memorează datele celui de al doilea soi. Pentru lalelele din fiecare soi se achită în total aceeași sumă, de câte 1000 de lei.
- Declarați corespunzător variabilele **cod1** și **cod2** și scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran, pentru fiecare soi, în ordine oarecare, pe linii separate, codul și numărul de fire achiziționate, ca în exemplu. Valorile afișate pe aceeași linie sunt separate printr-un spațiu.
- Exemplu:** dacă **cod1** memorează litera **A** și **pret1** memorează valoarea **5**, **cod2** memorează litera **P** și **pret2** memorează valoarea **4**, atunci se afișează pe ecran:
- A 200**
P 250
- (6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Un număr natural nenul, n , se numește număr **abundent** dacă $S(n) > n$, pentru orice număr natural nenul k ($k \leq n-1$), unde s-a notat cu $S(i)$ suma divizorilor pozitivi ai numărului natural nenul i . Se citește un număr natural, n ($n \geq 2$), și se cere să se scrie valoarea 1 , dacă n este un număr abundent, sau valoarea 0 , în caz contrar.
- Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.
- Exemplu:** pentru $n=6$, se scrie 1 ($S(6)/6=2$), iar cel mai mare raport obținut pentru valori strict mai mici decât 6 este $S(4)/4=1.75$, iar pentru $n=7$ sau $n=8$, se scrie 0 ($S(7)/7=1.14$, $S(8)/8=1.87$). (10p.)
2. Pentru a identifica punctele în care se concentrează apa în albia unui râu în cazul secetei, se determină talvegul acesteia – linia care unește punctele cele mai adânci ale albiei. În acest scop deocamdată s-au stabilit două secțiuni transversale pe cursul apei, și în cadrul fiecărei secțiuni s-a măsurat adâncimea apei în np puncte, numerotate începând de la 1 . Din fiecare secțiune, în ordine, se include în talveg cel mai adânc punct al acesteia, iar dacă în secțiune sunt mai multe puncte aflate la aceeași adâncime, maximă, se va lua în considerare doar primul dintre ele, ca în exemplu.
- Scrieți un program C/C++ care citește de la tastatură un număr natural, np ($np \in [1, 50]$), $2 \cdot np$ elemente ale unui tablou unidimensional, numere naturale din intervalul $[0, 10^2]$. Primele np valori corespund primei secțiuni, ultimele np valori corespund celei de a doua secțiuni, iar valorile memorate reprezintă adâncimile celor np puncte stabilite pentru acea secțiune, în ordinea numerotării lor. Programul afișează pe ecran, pentru fiecare secțiune, o pereche formată din numărul de ordine al secțiunii și numărul de ordine al punctului său care s-a inclus în talveg. Numerele din fiecare pereche sunt afișate separate prin câte un caracter : (două puncte), iar fiecare pereche este urmată de un spațiu.
- Exemplu:** pentru $np=4$ și tabloul $(\underline{2}, \underline{4}, \underline{5}, \underline{3}), (\underline{1}, \underline{3}, \underline{2}, \underline{3})$, se afișează pe ecran valorile:
- 1 : 3 2 : 2**
- (10p.)
3. Un număr natural x este numit prefix al unui număr natural y dacă este egal cu y sau se obține din acesta prin eliminarea a cel puțin unei cifre de la dreapta sa, și este numit sufix al lui y dacă este egal cu y sau dacă se obține din acesta prin eliminarea a cel puțin unei cifre de la stânga sa.
- Exemplu:** 15 este prefix pentru $\underline{154}$ sau $\underline{1521}$, și este sufix pentru $\underline{3415}$ sau $\underline{5115}$.
- Fișierul **bac.txt** conține maximum 10^6 numere naturale din intervalul $[10^2, 10^3]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran numărul valorilor de două cifre care apar de același număr de ori ca sufix, respectiv ca prefix al numerelor din șirul aflat în fișier. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.
- Exemplu:** dacă fișierul are conținutul
- 342 164 234 534 111 312 908 807 345 342 716 834 102 310
- se afișează pe ecran: **4** (pentru valorile **10**, **11**, **16**, **34**).
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)