

Examenul de bacalaureat național 2020

Proba E. d)
Informatică
Limbajul C/C++

Varianta 5

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 poate memora un număr natural din intervalul $[0, 10^9)$. Indicați valoarea maximă pe care o poate avea expresia C/C++ alăturată. **x*2020**
- a. 20.2 b. 2019 c. 2020 d. 495049
2. Tablourile unidimensionale **A** și **B** au elementele: **A**=(2, 22, 25, 32, 48), iar **B**=(3, 5, 8, 45, 60). Indicați succesiunea de valori care pot fi, în această ordine, elementele tabloului obținut în urma interclasării lui **A** și **B** în ordine crescătoare.
- a. (2, 5, 8, 32, 48) b. (2, 3, 5, 8, 22, 25, 32, 45, 48, 60)
- c. (2, 3, 5, 22, 8, 25, 32, 45, 48, 60) d. (2, 22, 25, 32, 48, 3, 5, 8, 45, 60)
3. Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea **x=0**, se aplică metoda căutării binare, iar succesiunea de elemente ale tabloului a căror valoare se compară cu valoarea lui **x** pe parcursul aplicării metodei indicate este: 20, 2, 0. Indicați succesiunea de valori care pot fi, în această ordine, elementele tabloului.
- a. (0, 1, 2, 4, 20, 40, 64) b. (0, 2, 4, 8, 16, 20, 32)
- c. (0, 2, 20, 0, 4, 40, 0, 8, 80) d. (0, 2, 4, 20, 30, 32, 40)
4. Variabilele **A**, **B** și **C** sunt de tip real și memorează coeficienții unei ecuații de gradul al II-lea **Ax²+Bx+C=0**. Indicați expresia C/C++ care calculează valoarea discriminantului ecuației (**B²-4·A·C**).
- a. **pow(B, 2)-4*A*C** b. **sqrt(B)-4*A*C**
- c. **pow(B*B-4*A*C, 2)** d. **sqrt(B*B-4*A*C)**
5. În secvența alăturată toate variabilele sunt de tip întreg și memorează numere naturale. Indicați expresia care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila **y** să memoreze valoarea 2ⁿ.
- ```
y=1;
x=2;
while(n>0)
 if(.....) { y=y*x; n=n-1; }
 else { x=x*x; n=n/2; }
```
- a. **n%2==0**                      b. **n/2==0**                      c. **n%2==1**                      d. **n/2!=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** și cu **[c]** partea întreagă a numărului real **c**.
- ```
citește n (număr natural)
m ← 0
repetă
    c ← n%10; n ← [n/10]
    dacă c < 5 atunci m ← m-2*c
    altfel m ← m+c
    ──
până când n = 0
dacă m = 0 atunci scrie 'DA'
altfel scrie m, 'NU'
    ──
```
- a. Scrieți ce se afișează dacă se citește numărul 247388. (6p.)
- b. Scrieți două numere din intervalul $[10^2, 10^3)$, cu cifre distincte, care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze **DA**. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **repetă... până când** cu o structură repetitivă cu test inițial. (6p.)

2. Pentru un procesor se memorează o literă mare a alfabetului englez, în funcție de producătorul acestuia: **I**, dacă producătorul are numele **Intel**, sau **A**, dacă producătorul are numele **Advanced Micro Devices**. Variabila **p1** memorează litera corespunzătoare producătorului unui procesor, iar variabila **p2** memorează litera corespunzătoare producătorului unui alt procesor. Declarați variabilele **p1** și **p2** și scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran numele producătorului, dacă ambele procesoare au același producător, sau mesajul **Producatori diferiti**, în caz contrar.
Exemplu: dacă atât **p1** cât și **p2** memorează litera **A**, în urma executării secvenței se afișează **Advanced Micro Devices** (6p.)
3. Variabilele **x1**, **x2** și **x3** sunt de tip întreg și memorează numere naturale distincte: una dintre ele un număr impar, iar celelalte două câte un număr par. Scrieți o secvență de instrucțiuni în urma executării căreia să se afișeze pe prima linie a ecranului valorile inițiale ale variabilelor **x1**, **x2** și **x3**, în această ordine, iar pe a doua linie a ecranului, aceleași valori, dar astfel încât cele pare să fie ordonate crescător, iar cea impară să își păstreze poziția. Numerele afișate pe aceeași linie a ecranului sunt separate prin câte un spațiu.
Exemplu: dacă **x1=8**, **x2=5**, **x3=4**, în urma executării secvenței se afișează
8 5 4
4 5 8 (6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Un număr natural nenul se numește **p-număr** dacă are aceeași paritate cu suma divizorilor săi pozitivi.
Exemplu: **10** și **25** sunt p-numere (**10** are aceeași paritate cu **18=1+2+5+10**, iar **25** are aceeași paritate cu **31=1+5+25**).
Se citesc trei numere naturale nenule, **a**, **b** și **k** ($a \leq b$), și se cere să se scrie cel de al **k**-lea p-număr din intervalul **[a, b]** sau **-1**, dacă nu există cel puțin **k** astfel de numere în acest interval. Scrieți în pseudocod algoritmul corespunzător.
Exemplu: dacă **a=27**, **b=50** și **k=3**, atunci se scrie **34**. (10p.)
2. Două numere distincte se numesc **în oglindă** dacă fiecare dintre ele se obține prin citirea cifrelor celuilalt de la dreapta la stânga.
Exemplu: **127** și **721** sunt în oglindă, iar pentru numerele **100** sau **121** nu există niciun număr cu care să fie în oglindă.
Scrieți un program C/C++ care citește de la tastatură un număr natural **n** ($n \in [2, 50]$), apoi cele **n** elemente ale unui tablou unidimensional, numere naturale din intervalul **[0, 10⁴)**. Programul transformă tabloul în memorie înlocuind fiecare element de trei cifre cu numărul cu care el este în oglindă, dacă acesta există, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, separate prin câte un spațiu, iar dacă nu s-a înlocuit niciun element se afișează pe ecran mesajul **nu exista**.
Exemplu: pentru **n=9** și tabloul (**721, 72, 6, 108, 180, 1221, 121, 537, 1000**), se obține tabloul (**127, 72, 6, 801, 180, 1221, 121, 735, 1000**).
iar pentru **n=4** și tabloul (**525, 72, 6, 120**), se afișează pe ecran mesajul **nu exista**. (10p.)
3. Fișierul **bac.txt** conține un șir de cel mult **10⁶** numere naturale din intervalul **[1, 10⁹)**, separate prin câte un spațiu.
Se cere să se afișeze pe ecran, separate printr-un spațiu, două numere naturale **a** și **b** ($a < b$), astfel încât oricare termen al șirului să aparțină intervalului **(a, b)**, iar valoarea expresiei **b-a** să fie minimă. Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al spațiului de memorie utilizat.
Exemplu: dacă fișierul conține valorile **7 2 40 5 10 15 11 122 350 18** se afișează pe ecran numerele **1 351**.
a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)