

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

Testul 16

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filieră vocațională, profil militar, specializare matematică-informatică

- 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.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

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. Expresia C/C++
$$(x < 17) \text{ || } !(x \leq 18 \text{ || } x \geq 20) \text{ || } (x > 21)$$
 are valoarea 0 pentru orice valoare a variabilei întregi x din mulțimea:
a. {17, 18, 19} b. {17, 18, 20, 21} c. {18, 19, 20} d. {18, 19, 20, 21}
2. Utilizând metoda backtracking sunt generate toate numerele din intervalul [100, 999] cu proprietatea că au cifrele în ordine crescătoare și cifrele aflate pe pozitii consecutive sunt de paritate diferită. Primele cinci soluții generate sunt, în această ordine, 123, 125, 127, 129, 145. Indicați cel de al 9-lea număr generat.
a. 149 b. 167 c. 169 d. 189
3. Fiecare dintre variabilele A și B , declarate alături, memorează coordonatele (x abscisa, iar y ordonata) câte unui punct în sistemul de coordonate xy . Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă segmentul cu extremitățile în punctele corespunzătoare variabilelor A și B intersectează axa ox a sistemului de coordonate.
a. $(A-y) * (B-y) \leq 0$ b. $A.y * B.y \leq 0$ c. $\text{punct}.y(A, B) \leq 0$ d. $y.A * y.B \leq 0$


```
struct punct
{
    int x, y;
} A, B;
```
4. Un graf orientat cu 6 vârfuri, numerotate de la 1 la 6, are arcele (1, 2), (1, 6), (1, 5), (2, 3), (3, 1), (3, 5), (4, 6), (5, 6), (6, 2). Indicați numărul de vârfuri care au gradul extern mai mare decât gradul intern.
a. 1 b. 2 c. 3 d. 4
5. Un graf neorientat are 50 de noduri și 32 de muchii. Indicați numărul maxim de componente conexe pe care le poate avea graful.
a. 25 b. 31 c. 33 d. 42

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

S-a notat cu $a \leq b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .

- a. Scrieți numărul afișat în urma executării algoritmului dacă se citesc, în această ordine, numerele 812302105 și 4. (6p.)

- b. Dacă pentru k se citește numărul 1, scrieți trei numere din intervalul $[10^3, 10^4]$ care pot fi citite pentru n , astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze un număr format din două cifre identice. (6p.)

- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)

- d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adevarat structura **repetă...până când** cu o structură repetitivă de alt tip. (6p.)

```

citeză n,k
(numere naturale)
dacă k=0 atunci nr←-1
altfel
  nr←0
  p←1
  repetă
    c←n%10; n←[n/10]
    dacă c%2=0 atunci
      nr←nr+c*p; p←p*10
    altfel k←k-1
  până când n=0 sau k=0
scrie nr
  
```

2. Subprogramul f este definit

alăturat. Scrieți ce se afișează în urma apelului de mai jos.

$f(5);$ (6p.)

```

void f(int n)
{ if (n!=0)
  { if (n%2==1) cout<<n<<' '; | printf("%d ",n);
    f(n-1);
    cout<<n<<' '; | printf("%d ",n);
  }
  else cout<<endl; | printf("\n");
}
  
```

3. Variabilele $s1$ și $s2$ pot memora câte un

șir cu cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței (6p.)

```

strcpy(s1,"bacalaureat2020");
cout<<strlen(s1); | printf("%d",strlen(s1));
strcpy(s2,s1+11); strcpy(s1+3,s2);
cout<<s1; | printf("%s",s1);
  
```

SUBIECTUL al III-lea (30 de puncte)

- 1. Subprogramul $nrDivPrimi$ are un singur parametru, n , prin care primește un număr natural ($n \in [2, 10^9]$). Subprogramul returnează numărul divizorilor care, în descompunerea în factori primi a lui n , apar la o putere impară.**

Scrieți definiția completă a subprogramului.

Exemplu: dacă $n=9000$, subprogramul returnează 2 ($9000=2^3 \cdot 3^2 \cdot 5^3$). (10p.)

- 2. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul $[2, 10^2]$, n și m , și construiește în memorie un tablou bidimensional cu n linii și m coloane, cu proprietatea că parcurgându-l linie cu linie de sus în jos și fiecare linie de la stânga la dreapta, se obține șirul primelor $n \cdot m$ pătrate perfecte impare, ordonat strict descrescător, ca în exemplu.**

Elementele tabloului obținut se afișează pe ecran, fiecare linie a tabloului pe câte o linie a ecranului, valorile de pe aceeași linie fiind separate prin câte un spațiu. 121 81 49
25 9 1

Exemplu: pentru $n=2$, $m=3$ se obține tabloul alăturat. (10p.)

- 3. Fișierul $bac.in$ conține numere naturale: pe prima linie două numere din intervalul $[1, 10^6]$, m și n , pe a doua linie un șir de m numere din intervalul $[1, 10^9]$, iar pe a treia linie un șir de n numere din intervalul $[1, 10^9]$. Numerele aflate pe aceeași linie a fișierului sunt separate prin câte un spațiu, și ambele șiruri sunt ordonate crescător.**

Se cere să se afișeze pe ecran, în ordine strict crescătoare, un șir format dintr-un număr maxim de termeni care aparțin cel puțin unuia dintre cele două șiruri, astfel încât oricare două elemente aflate pe poziții consecutive să fie de paritate diferită. Numerele afișate sunt separate prin câte un spațiu.

Proiectați un algoritm eficient din punctul de vedere al timpului de execuțare.

Exemplu: dacă fișierul are conținutul alăturat, se afișează pe ecran

2 3 4 5 8 11 14 sau 2 3 4 5 10 11 14

8 5
2 4 5 8 8 11 14 14
3 4 5 5 10

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

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