

Examenul național de bacalaureat 2021

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

Model

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

- 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.
- 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. Indicați valoarea expresiei C/C++ alăturate. `21/2*2-5`
a. 0 b. 0.25 c. 15 d. 16

2. Variabila `x` este declarată alăturat. Indicați secvența care, în urma executării, afișează pe ecran ziua, luna și anul corespunzătoare unei date calendaristice memorate în variabila `x`. `struct data`
`{`
`int zi, luna, an;`
`}x;`
a. `cout<<x.zi<<" "<<x.luna<<" "<<x.an; | printf("%d %d %d",x.zi,x.luna,x.an);`
b. `cout<<zi.x<<" "<<luna.x<<" "<<an.x; | printf("%d %d %d",zi.x,luna.x,an.x);`
c. `cout<<x(zi," ",luna," ",an); | printf("%d %d %d",x(zi,luna,an));`
d. `cout<<data.x.zi<<" "<<data.x.luna<<" "<<data.x.an;`
`| printf("%d %d %d",data.x.zi,data.x.luna,data.x.an);`

3. Subprogramul `f` este definit alăturat. Indicați valoarea `f(3,1)`. `int f(int n,int x)`
`{ if(n!=0)`
`{ x=x+1;`
`return f(n-1,x)-x;`
`}`
`return 0;`
`}`
a. 0 b. -3 c. -6 d. -9

4. Un arbore cu rădăcină are 8 noduri, numerotate de la 1 la 8, și muchiile `[1,3]`, `[1,7]`, `[1,8]`, `[2,4]`, `[3,5]`, `[3,6]`, `[4,5]`. Indicați numărul maxim de frunze ale arborelui, în funcție de rădăcina aleasă.
a. 5 b. 4 c. 3 d. 2

5. Un graf neorientat complet are 21 de noduri. Indicați numărul de muchii ce pot fi eliminate, astfel încât graful parțial obținut să fie conex și fără cicluri.
a. 211 b. 209 c. 190 d. 188

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 numărul afișat în urma executării algoritmului dacă se citește, în această ordine, numerele 3845267 și 3. (6p.)
- b. Dacă pentru variabila k se citește 2, 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 20. (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.)
- ```

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

```
2. Utilizând metoda backtracking se generează toate numerele din intervalul  $[10^4, 10^5)$ , cu cifrele în ordine strict crescătoare. Primele trei numere generate sunt 12345, 12346, 12347. Scrieți, în ordinea obținerii, ultimele trei numere generate care au prima cifră 4. (6p.)
3. Variabila  $s$  memorează un șir cu cel mult 20 de caractere, iar celelalte variabile sunt de tip întreg. Scrieți ce se afișează pe ecran în urma executării secvenței alăturate. (6p.)
- ```

k = 'a' - 'A';
strcpy(s, "A realizat tot");
for(i=strlen(s)-1; i>=0; i--)
    if(strchr("aeiou", s[i]) != NULL)
        { s[i] = s[i] - k;
          cout << s[i]; | printf("%c", s[i]);
        }
    
```

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul **prime** are trei parametri:
- n , prin care primește un număr natural ($n \in [4, 10^9]$);
 - x și y , prin care furnizează cele mai mari două numere prime din intervalul $[1, n)$, $x < y$.
- Scrieți definiția completă a subprogramului.
Exemplu: dacă $n=49$, în urma apelului $x=43$, $y=47$. (10p.)
2. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul $[2, 10^2)$, m și n , și construiește în memorie un tablou bidimensional cu m linii și n 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 $m \cdot n$ **pătrate perfecte pare**, 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.
Exemplu: pentru $m=2$, $n=3$ se obține tabloul alăturat. (10p.)
- | | | |
|-----|----|----|
| 100 | 64 | 36 |
| 16 | 4 | 0 |
3. Fișierul **cheltuieli.in** are cel mult 10^6 linii, fiecare linie conținând câte trei numere naturale din intervalul $[1, 10^2]$, reprezentând, în această ordine, date despre câte o achiziție: tipul produsului cumpărat, numărul de produse de acest tip cumpărate, respectiv prețul unui astfel de produs la acel moment. Numerele aflate pe aceeași linie sunt separate prin câte un spațiu.
Se cere să se afișeze pe ecran cea mai mare sumă cheltuită pentru toate produsele de același tip, precum și numărul de tipuri de produse pentru care s-a obținut această sumă. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.
Exemplu: dacă fișierul **cheltuieli.in** are conținutul alăturat, se afișează pe ecran: 26 2
(s-a cheltuit suma maximă 26 pentru produsele de tipul 1 și 4: $26 = 16 \cdot 1 + 5 \cdot 2 = 1 \cdot 10 + 2 \cdot 8$)
- | | | |
|---|----|----|
| 4 | 1 | 10 |
| 1 | 16 | 1 |
| 4 | 2 | 8 |
| 2 | 1 | 5 |
| 1 | 5 | 2 |
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)