

Examenul național de bacalaureat 2021

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

Testul 5

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 muchie are extremități distincte și oricare două 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.

- Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numerele naturale memorate în variabilele întregi x și y sunt pare, consecutive.
 - $(x-y==2) \ \&\& \ (y-x==2)$
 - $(x==2) \ \&\& \ (y==4)$
 - $x-y==2$
 - $((x-y==2) \ || \ (y-x==2)) \ \&\& \ (x\%2==0)$
- Subprogramul `f` este definit alăturat. Indicați ce se afișează în urma apelului de mai jos.
`f(7552021, 1);`

```
void f (int n, int k)
{ if (n!=0)
  { f(n/10,k+1);
    if(n%10==k) cout<<k; | printf("%d",k);
  }
}
```

 - 7521
 - 1257
 - 21
 - 1
- Utilizând metoda backtracking se generează toate posibilitățile de a scrie numărul 10 ca sumă de numere prime. Soluțiile generate sunt, în această ordine: 2+2+2+2+2, 2+2+3+3, 2+3+5, 3+7, 5+5. Folosind aceeași metodă se generează toate posibilitățile de a scrie numărul 9 ca sumă de numere prime. Indicați a treia soluție generată.
 - 2+2+2+3
 - 2+2+5
 - 2+7
 - 3+3+3
- Un arbore cu rădăcină, cu 9 noduri, numerotate de la 1 la 9, este reprezentat prin vectorul de „tați” (8, 7, 6, 5, 7, 7, 8, 0, 8). Indicați toți descendenții nodului 7.
 - 2, 5, 6
 - 2, 3, 5, 6
 - 2, 4, 5, 6
 - 2, 3, 4, 5, 6
- Un graf neorientat are 6 noduri, numerotate de la 1 la 6, și muchiile [1, 2], [1, 3], [2, 3], [2, 4], [2, 5], [2, 6], [3, 4], [4, 5]. Indicați numărul nodurilor care au gradul un număr impar.
 - 5
 - 4
 - 3
 - 2

SUBIECTUL al II-lea (40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.**
S-a notat cu `[c]` partea întreagă a numărului real `c`.
 - Scrieți valoarea afișată dacă se citesc, în această ordine, numerele 12, 7, 354, 9, 630, 0. (6p.)
 - Scrieți o secvență de numere din intervalul $[0, 10^4)$ care pot fi citite, în această ordine, astfel încât, în urma executării algoritmului, să se afișeze 321. (6p.)
 - Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
 - Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat a doua structură `repetă...până când` cu o structură repetitivă cu test inițial. (6p.)

```
citește x
  (număr natural nenul)
y ← 0
repetă
  dacă x > 9 atunci
    repetă
      x ← [x/10]
    până când x ≤ 9
  y ← y*10+x
citește x (număr natural)
până când x=0
scrie y
```

2. Variabila `c` memorează simultan următoarele date despre o carte dintr-o bibliotecă: titlul (un șir de maximum 50 de caractere), numele autorului/autorilor și numărul de exemplare (maximum 10^2); o carte poate avea cel mult 10 autori, fiecare având un nume de maximum 50 de caractere. Știind că expresiile C/C++ de mai jos au ca valori două șiruri de caractere, reprezentând titlul, respectiv numele primului autor al cărții, și un număr natural reprezentând numărul de exemplare ale acestei cărți, scrieți definiția unei structuri cu eticheta `carte`, care permite memorarea datelor despre o carte, și declarați corespunzător variabila `c`.

`c.titlu` `c.autor[0]` `c.nrExemplare` (6p.)

3. Variabila `i` este de tip întreg, iar variabila `s` permite memorarea unui șir cu cel mult 10^2 caractere. Scrieți ce se afișează pe ecran în urma executării secvenței alăturate. (6p.)

```
strcpy(s, "informatica");
cout<<strlen(s); | printf("%d", strlen(s));
for (i=0; i<strlen(s); i++)
    if (strchr("aeiou", s[i])!=NULL)
        s[i]= '*';
cout<<s; | printf("%s", s);
```

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul `identice` are un singur parametru, `n`, prin care primește un număr natural ($n \in [10, 10^9]$). Subprogramul returnează valoarea 1, dacă numărul `n` are toate cifrele egale, sau valoarea 0 în caz contrar. Scrieți definiția completă a subprogramului.

Exemplu: dacă `n=2222`, subprogramul returnează valoarea 1, iar dacă `n=212`, subprogramul returnează valoarea 0. (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, numerotate începând cu 1, astfel încât elementul de pe linia `i` și coloana `j` primește ca valoare ultima cifră a produsului `i·j`.

Programul afișează pe ecran elementele tabloului obținut, linie cu linie, fiecare linie a tabloului pe câte o linie a ecranului, cu valorile aflate pe aceeași linie separate prin câte un spațiu.

Exemplu: dacă `m=4` și `n=5` se afișează pe ecran tabloul alăturat. (10p.)

1	2	3	4	5
2	4	6	8	0
3	6	9	2	5
4	8	2	6	0

3. Fișierul `bac.txt` conține numere naturale din intervalul $[2, 10^6]$: pe prima linie `n`, iar pe a doua linie un șir de `n` numere, separate prin câte un spațiu.

Se cere să se afișeze pe ecran, pentru fiecare număr natural `i` ($i \in [1, n]$), cea mai mare dintre primele `i` valori ale șirului aflat în fișier. Numerele afișate sunt separate prin câte un spațiu. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

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

12	4	6	6	7	8	8	8	8	9	10	10
4	6	3	7	8	1	6	2	7	9	10	8

- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)