



Simulare pentru EXAMENUL DE BACALAUREAT - ianuarie 2023
Proba E.d)
INFORMATICĂ

Varianta 1

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.

- Variabila x este de tip real. O expresie C/C++ care are ca valoare partea întreagă a radicalului de ordin 3 din x este:
 - `int(pow(x,0.3))`
 - `int(pow(x,3))`
 - `int(pow(x,1./3))`
 - `int(pow(x,1/3))`
- Subprogramul `fac` este definit alături și determină valoarea lui $n!$, pentru orice n număr natural, returnând rezultatul prin parametrul f . Care va fi secvența cu care se vor înlocui punctele de suspensie astfel încât subprogramul să funcționeze corect.

<pre>void fac(int n ,int & f) { if(n == 0) f = 1; else { int g; f = g * n; } }</pre>	<pre>void fac(int n ,int & f) { if(n == 0) f = 1; else { int g; f = g * n; } }</pre>
--	--

 - `fac(n,g-1);`
 - `fac(n,g);`
 - `fac(n-1,g-1);`
 - `fac(n-1,g);`
- Variabila s este de tip șir de caractere, iar variabilele $c1$ și $c2$ sunt de tip char. Care expresie are valoarea 1 dacă și numai dacă șirul de caractere s conține caracterele memorate de variabilele $c1$ și $c2$?
 - `strstr(s,c1+c2)!=0`
 - `strchr(s,c1)!=0 || strchr(s,c2)!=0`
 - `strchr(strchr(s,c1),c2)!=0`
 - `strchr(s,c1) && strchr(s,c2)`
- Utilizând metoda backtracking, se generează toate subiectele care cuprind patru capitole distincte de programare din mulțimea {**cifre**, **divizori**, **matrici**, **pointeri**, **secvente**, **siruri**}, fiecare subiect respectând următoarele condiții: dacă există **cifre**, este primul capitol, dacă există **divizori**, este ultimul capitol, nu sunt subiecte cu **pointeri** și **siruri** în același subiect. Două subiecte sunt distincte dacă diferă prin cel puțin un capitol sau prin ordinea acestora. Primele trei subiecte generate sunt, în această ordine: (**cifre,matrici,pointeri,divizori**), (**cifre,matrici,pointeri,secvente**), (**cifre,matrici,secvente,divizori**), Indicați ultimul subiect generat.
 - matrici,pointeri,secvente,siruri**
 - siruri,secvente,pointeri,matrici**
 - siruri,secvente,matrici,divizori**
 - siruri,secvente,matrici,cifre**
- Într-un arbore cu rădăcină avem 2021 noduri. Fiecare nivel are un număr pătrat perfect de noduri. Toate frunzele sunt pe același nivel. Care este numărul minim de niveluri pentru a distribui toate cele 2021 de noduri ale arborelui?
 - 2
 - 3
 - 4
 - 5

SUBIECTUL al II-lea (40 de puncte)

- Se consideră declarațiile alăturate: Structura `fracție` memorează o fracție prin numărător și numitor. Declarați variabilele necesare și scrieți o secvență de instrucțiuni care va determina în variabila `suma` o fracție ireductibilă reprezentând suma fracțiilor `f1` și `f2`. (6p.)

<pre>struct fracție { int numarator, numitor; };</pre>	<pre>struct fracție f1,f2, suma;</pre>
--	--



2. Algoritmul alăturat este reprezentat în pseudocod. S-a notat cu $a \div b$ restul împărțirii numărului natural a la numărul natural nenul b și cu $[c]$ partea întreagă a numărului real c .
- Scrieți numărul afișat dacă se citesc valorile: 3, 3, 2018, 2019, 2020. (6p.)
 - Dacă pentru variabilele n și c se citesc valorile 6 și 5, scrieți un șir valori ce urmează a fi citite astfel încât valoarea afișată să fie 6. (6p.)
 - Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
 - Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască structurile repetitive **repetă...până când** cu câte o structură repetitivă **cât_timp ... execută**. (6p.)
3. Se consideră graful orientat cu vârfurile numerotate de la 1 la 5 și arcele: (1,5), (2,1), (2,3), (3,1), (3,2), (3,5), (4,1), (4,2), (4,3), (5,1). Care este numărul de componente tare conexe? Să se determine numărul minim de arce ce trebuie adăugate astfel încât graful să devină tare conex. (6p.)

```

citește n, c (numere naturale nenule)
m←0, i←0
repetă
    citește x (număr natural nenul)
    i←i+1
    repetă
        s←0
        repetă
            s←s+x%10, x←[x/10]
        până când x=0
        x←s
    până când x<10
    dacă x=c atunci
        m←m+1
    până când i=n
scrie m
    
```

SUBIECTUL al III-lea

(30 de puncte)

- Subprogramul **ecran**, are un singur parametru:
 - n , prin care primește un număr natural ($n \in [1, 10^6]$). Subprogramul afișează pe ecran prefixele și sufixele numărului sub formă triunghiulară, delimitate de steluțe, ca în exemplul următor (unde $n=2023$):


```

2023*2023
202***023
20*****23
2*****3
*****
                    
```

 Scrieți definiția completă a subprogramului. (10p.)
- Se dă o matrice cu n linii și m coloane și elemente numere naturale distincte două câte două. Numim **transformare** eliminarea din matrice a liniei și coloanei pe care se află elementul maxim, respectiv a liniei și coloanei pe care se află elementul minim. Programul citește de la tastatură numerele n și m , (având maximum 2 cifre fiecare) iar apoi $n*m$ numere naturale de maximum 6 cifre fiecare, separate prin spații, reprezentând elementele matricei, linie cu linie. Programul afișează pe ecran pe prima linie noile dimensiuni ale matricii **transformate** în memorie, iar apoi pe liniile următoare elementele matricei, câte o linie a matricei pe o linie a ecranului, elementele de pe o linie fiind separate prin câte un spațiu.

Date de intrare	Date de ieșire
3 3	1 1
1 2 3	5
4 5 6	
7 8 9	

 (10p.)
- Se consideră un șir s format după regula alăturată, unde s-a notat cu $a \odot b$ numărul obținut prin concatenarea cifrelor lui a și b , în această ordine. Exemplu: pentru $x=5$ se obține șirul: 5, 6, 65, 656, 65665, 65665656,

$$s_n = \begin{cases} x, & \text{dacă } n = 1 \\ x + 1, & \text{dacă } n = 2 \\ s_{n-1} \odot s_{n-2}, & \text{dacă } n > 2. \end{cases}$$

Fișierul text **bac.in**, conține pe prima linie două numere x ($1 \leq x \leq 50$) și k ($1 \leq k \leq 100000$), separate printr-un spațiu, iar pe a doua linie un număr format din exact k cifre (caractere fără spații), reprezentând un termen al șirului s , diferit de x . Afișați pe ecran termenul din șirul s , care este generat înaintea numărului de pe linia a doua a fișierului **bac.in**. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fișierul conține valorile alăturate, se va afișa pe ecran numărul 656.

5 5
65665

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