

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.
- Identifierii utilizati 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.

1. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă numărul memorat în variabila întreagă **x** aparține reuniunii de intervale $[-2021, -2020] \cup [2020, 2021]$.
 - a. $x >= -2021 \ || \ x <= -2020 \ || \ x >= 2020 \ || \ x <= 2021$
 - b. $x >= -2021 \ \&\& \ x <= -2020 \ \&\& \ x >= 2020 \ \&\& \ x <= 2021$
 - c. $! (x < -2021 \ || \ x > -2020) \ || \ ! (x < 2020 \ || \ x > 2021)$
 - d. $! (x < -2021 \ \&\& \ x > 2021 \ \&\& \ x > -2020 \ || \ x < 2020)$
2. Variabila **x** este declarată alăturat. Indicați o instrucțiune de atribuire corectă din punctul de vedere sintactic.

```
struct masina
{
    char serie;
    int rating;
}x;
```

 - a. **x.rating=x.serie-'A'**
 - b. **x.masina.rating=2021;**
 - c. **x(serie)='A'**
 - d. **x=(2021,'A',10);**
3. Utilizând metoda backtracking se generază toate posibilitățile de a forma șiruri din câte 3 mărgele de culori distincte din mulțimea {roșu, galben, verde, albastru, violet}. Două șiruri sunt distincte dacă diferă prin cel puțin o culoare a mărgelelor sau prin ordinea acestora. Primele patru soluții generate sunt, în această ordine: (roșu, galben, verde), (roșu, galben, albastru), (roșu, galben, violet), (roșu, verde, galben). Indicați a zecea soluție generată.
 - a. (galben, roșu, verde)
 - b. (roșu, albastru, violet)
 - c. (roșu, violet, verde)
 - d. (roșu, violet, galben)
4. Subprogramul **f** este definit alăturat, parțial. Indicați o expresie care poate înlocui punctele de suspensie astfel încât, $f(2021, 1880)$ să aibă valoarea celui mai mare divizor comun al numerelor 2021 și 1880.

```
int f (int x, int y)
{
    if(y==0) return x;
    else return.....;
}
```

 - a. **f(x,y%y)**
 - b. **f(y,x%y);**
 - c. **f(x%y,y)**
 - d. **f(y%y,x)**
5. Un arbore are 5 noduri, numerotate de la 1 la 5. Indicați o succesiune de valori care pot reprezenta gradele nodurilor unui astfel de arbore.
 - a. 0,1,1,1,1,5
 - b. 1,1,1,1,3
 - c. 1,1,2,2,2
 - d. 1,1,2,2,3

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

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

a. Scrieți ce se afișează dacă se citesc, în această ordine, numerele 5 și 2. **(6p.)**

b. Dacă pentru variabila k se citește valoarea 6, scrieți cea mai mică și cea mai mare valoare din intervalul $[1, 30]$ care pot fi citite pentru variabila n astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, ultimul număr afișat să fie 12. **(6p.)**

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

d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adevarat ultima structură **pentru...execută** cu o structură repetitivă de alt tip. **(6p.)**

2. Un graf neorientat cu 6 noduri, numerotate de la 1 la 6, este reprezentat prin matricea de adiacență alăturată. Pentru acest graf scrieți un lanț elementar, precum și un ciclu care nu este elementar. **(6p.)**

0	1	0	0	0	1
1	0	0	1	0	1
0	0	0	0	1	1
0	1	0	0	1	0
0	0	1	1	0	1
1	1	1	0	1	0

3. Variabila i este de tip întreg, iar variabila s permite memorarea unui sir cu cel mult 50 de caractere, numai litere mari ale alfabetului englez. Scrieți secvența de mai jos înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran toate consoanele din sir, iar în locul vocalelor din mulțimea $\{O, A, U\}$ să se afișeze simbolul *.

Exemplu: dacă sirul este **CALCULATOARE**, se afișează ***E***L***U***RE** **(6p.)**

```
for(i=0;i<strlen(s);i++)
    if(.....) cout<<...; | printf(...);
    else cout<<...; | printf(...);
```

SUBIECTUL al III-lea

(30 de puncte)

1. Un număr y este numit **frate mai mare al unui număr x dacă x și y au același număr de cifre și fiecare cifră a lui y se poate obține din cifra aflată pe aceeași poziție în x adunând la aceasta valoarea 1.**

Subprogramul **frate** are doi parametri:

- x , prin care primește un număr natural ($x \in [0, 10^9]$);
- y , prin care furnizează fratele mai mare al lui x , sau -1, dacă nu se poate obține un astfel de număr.

Scrieți definiția completă a subprogramului.

Exemplu: dacă $x=1027$, după apel $y=2138$, iar dacă $x=9027$, după apel $y=-1$. **(10p.)**

2. Scrieți un program C/C++ care citește de la tastatură numere naturale: n ($n \in [2, 20]$), k ($k \in [1, n]$) și $n \cdot n$ numere din intervalul $[0, 10^9]$, elemente ale unui tablou bidimensional cu n linii și n coloane. Programul transformă tabloul în memorie, deplasând circular spre dreapta, cu câte o poziție, toate elementele situate pe linia a k -a, în stânga diagonalei secundare, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, fiecare linie pe câte o linie a ecranului, cu elementele fiecărei linii separate prin câte un spațiu.

Exemplu: pentru $n=5$, $k=2$ și tabloul

2	3	4	5	6
2	4	6	8	0
7	8	9	0	1
3	5	7	9	1
7	3	8	5	6

se obține tabloul

2	3	4	5	6
6	2	4	8	0
7	8	9	0	1
3	5	7	9	1
7	3	8	5	6

(10p.)

3. Fișierul **bac.txt conține un sir de cel mult 10^5 numere naturale din intervalul $[1, 10^9]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran cea mai mare poziție pe care ar putea-o ocupa primul termen al sirului aflat în fișier în sirul format cu aceleași valori, ordonat descrescător. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.**

Exemplu: dacă fișierul **bac.txt** conține numerele 15 7 15 17 6 4 21

se afișează pe ecran 4 (15 se află pe a treia și pe a patra poziție în sirul 21, 17, 15, 15, 7, 6, 4).

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