

Inspectoratul Școlar Județean Brăila
Simulare pentru EXAMENUL DE BACALAUREAT - decembrie 2025
Proba E.d) INFORMATICĂ

Filieră teoretică, profil real, specializare matematică-informatică / matematică -informatică -intensiv 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. Stabiliți care dintre următoarele expresii C/C++ are valoarea 1 dacă și numai dacă numărul întreg memorat în variabila ***z*** nu aparține intervalului $(-20, 20) \cup [35, 150]$.

- a. $(z < -20) \vee ((z \leq 34) \vee (z \geq 20)) \vee (z > 150)$
- b. $(z < -20) \vee ((z \leq 35) \wedge (z \geq 20)) \vee (z \geq 150)$
- c. $(z < -20) \vee ((z < 34) \wedge (z > 20)) \vee (z > 150)$
- d. $(z \leq -20) \vee ((z \leq 34) \wedge (z \geq 20)) \vee (z > 150)$

2. Subprogramul ***f*** este definit alăturat. Indicați ce se afișează în urma apelului de mai jos.
f(4);

```
void f(int n)
{if (n>0)
 {f(n-1); cout<<n<<' ' ;f(n-2); }
}
```

- a. 1 2 3 1 4 1 2 b. 1 2 3 4 1 c. 1 2 1 3 1 2 1 d. 0 1 2 3 0

3. La un **festival de știință**, organizatorii trebuie să stabilească ordinea a **trei ateliere** alese din mulțimea: robotică, astronomie, chimie, biologie, programare. Fiecare atelier aparține uneia dintre următoarele **categorii tematice**: -tehnologic: *robotică, programare*; -științific: *astronomie*; -experimental: *chimie, biologie*.

Se dorește generarea tuturor aranjamentelor ordonate lexicografic de câte 3 ateliere distincte, cu respectarea următoarelor **restricții**: nu pot fi alese două ateliere din aceeași categorie, atelierul *chimie* nu poate fi primul, atelierul *astronomie* trebuie să apară înaintea oricărui atelier tehnologic.

Primele patru seturi generate, în această ordine, sunt: (***astronomie, biologie, programare***), (***astronomie, biologie, robotică***), (***astronomie, chimie, programare***) și (***astronomie, chimie, robotică***). Care este **ultimul** set generat?

- a. (***robotica, chimie, astronomie***) b. (***biologie, astronomie, programare***)
- c. (***biologie, astronomie, robotică***) d. (***chimie, programare, astronomie***)

4. Fie un graf neorientat ***G*** cu **2026** noduri numerotate cu numere naturale consecutive de la 1 la 2026. Știind că oricare două noduri de aceeași paritate sunt adiacente, indicați cum trebuie modificat graful, astfel încât acesta să devină eulerian.

- a. se elimină două muchii b. se elimină o muchie și se adaugă două muchii noi
- c. se adaugă două muchii noi d. se elimină o muchie

5. Indicați câte componente tare conexe are graful orientat ***G*** = (***V***, ***E***) unde ***V*** = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}, iar ***E*** = {(1,2), (1,7), (1,10), (2,6), (3,2), (3,5), (3,9), (4,3), (4,6), (5,2), (6,1), (8,7), (9,6), (9,8)}.

- a. 5 b. 6 c. 7 d. 8

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod. S-a notat cu ***x%y*** restul împărțirii numărului natural ***x*** la numărul natural ***y***.

- a. Ce se afișează în urma rulării programului dacă se introduce șirul de valori 17 20 153 169 14 130 1250 1555 535 616 0. (6p.)
- b. Scrieți un șir de valori pentru variabila ***a*** astfel încât programul să afișeze în urma executării, mesajul 'NU'. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm echivalent

```
n←0
repetă
citește a (a număr natural)
dacă a!=0 atunci
    dacă a%2=0 atunci n←n+1
    altfel n←n-1
    ■
până când a=0
dacă n=0 atunci scrie 'DA'
altfel scrie 'NU'
■
```

Inspectoratul Școlar Județean Brăila

cu cel dat, înlocuind adecvat
structura **repetă...până când** cu o structură
repetitivă cu test inițial. (6p.)

2. Se consideră un arbore cu rădăcină reprezentat în memorie cu ajutorul vectorului de tați: **T = (2, 3, 0, 3, 3, 2, 6, 6, 4, 9)**. Scrieți care dintre nodurile arborelui sunt extremitățile finale ale unor lanțuri elementare de lungime 3 care au ca extremitate inițială rădăcina arborelui. (6p.)

3. Variabilele **sir1**, **sir2** și **rezultat** permit accesul la câte un șir de maximum 20 de caractere. Inițial, șirul accesat prin **rezultat** este vid. Șirul accesat prin **sir1** reprezintă o adresă URL simplă (fără **http://** sau **https://**), iar șirul accesat prin **sir2** reprezintă un prefix de domeniu (de exemplu, **"www."** sau un șir vid). Scrieți o secvență de instrucțiuni **C/C++** astfel încât, în urma executării acesteia, șirul accesat prin **rezultat** să reprezinte adresa URL completă, formată prin concatenarea prefixului de domeniu cu adresa URL simplă. Declarați corespunzător eventualele alte variabile utilizate. **Exemplu:** dacă prin variabila **sir1** se accesează șirul **"google.com"**, iar prin variabila **sir2** se accesează șirul **"www."**, atunci șirul format local este **"www.google.com"**, iar dacă prin variabila **sir1** se accesează șirul **"pbinfo.ro"**, iar prin variabila **sir2** se accesează șirul vid, atunci șirul format local este **"pbinfo.ro"**. (6p.)

Subiectul III

(30 de puncte)

1. Subprogramul **nrdiv** cu doi parametri întoarce numărul de divizori naturali ai unui număr natural **n** primit ca parametru și furnizează prin intermediul parametrului **d** cel mai mare divizor prim al lui **n**. Scrieți definiția completă a subprogramului **C/C++**. (10p.)

Exemplu: dacă **n=12**, după apel va returna numărul de divizori 6, iar **d** va fi 3

2. Scrieți un program **C/C++** care citește de la tastatură 2 numere naturale **n** și **m** din intervalul **[2,100]**, și construiește în memorie un tablou bidimensional cu **n** linii, numerotate de la 1 la **n** și **m** coloane, numerotate de la 1 la **m**, respectând următoarele reguli: elementele de pe prima linie sunt egale cu numărul coloanei pe care se află, elementele de pe celelalte linii sunt egale cu suma vecinilor elementului situați pe linia anterioară, ca în exemplu. Matricea construită se va afișa pe ecran, începând din colțul din dreapta jos, câte o linie a tabloului pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu. (10p.)

Exemplu:

n=3 m=4

Matricea inițială

1 2 3 4
3 6 9 7
9 18 22 16

Se afișează

16 22 18 9
7 9 6 3
4 3 2 1

3. Fișierul **bac.in** conține cel puțin 3 și cel mult 1000000000 de numere naturale mai mici sau egale cu 10000. Să se găsească și să afișeze pe ecran primul triplet de numere aflate pe poziții consecutive în fișier cu proprietatea că al doilea număr este egal cu media geometrică a celorlalte două valori din triplet. Dacă nu există un astfel de triplet se va afișa mesajul **"Nu există"**. Utilizați un algoritm eficient din punct de vedere al timpului de executare și a memoriei utilizate.

Exemplu

bac.in

5 7 3 2 10 50 13 3 9 27 4 10 5

Se afișează pe ecran

2 10 50

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

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