

Simulare pentru Examenul de bacalaureat ianuarie 2026

Proba E. d)
Informatică

Simulare 2

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)

1. Se consideră variabilele:

`int a = 7, b = 3, c = 5;`

`float x = 2.5, y = -1.0;`

Indicați rezultatul expresiei: $(x + a \% c) / (\text{float})(b + y) + a / b * c$

- a. 11 b. 11.75 c. 12 d. 12.25

2. Se consideră următoarea funcție recursivă:

```
int s(int t)
```

```
{
```

```
    if (t == 1) return 0;
```

```
    else if (t == 2) return 1;
```

```
    else return s(t-2) + s(t-1);
```

```
}
```

Dacă apelul inițial nu se ia în considerare, precizați câte autoapeluri vor fi făcute pentru apelul `s(6)`.

- a. 8 b. 10 c. 12 d. 14

3. Utilizând metoda backtracking se generează toate codurile formate din cinci caractere distincte ale mulțimii $\{a, b, c, d, e, f\}$. Primele cinci soluții generate sunt: `abcde`, `abcdf`, `abced`, `abcef`, `abcfed`. Indicați care sunt codurile generate imediat în fața soluției `dcbae`, și imediat după aceasta.

- a. `dcaef`; `dcafe` b. `dcafe`; `dcbafe` c. `dcbafe`; `dcbea` d. `dcbfed`; `dcbafe`

4. Numărul maxim de muchii dintr-un graf neorientat cu 16 noduri și 7 componente conexe este:

- a. 25 b. 36 c. 45 d. 55

5. Se consideră un arbore cu 10 noduri, numerotate de la 1 la 10 având vectorul de tați următor (0, 1, 1, 1, 1, 3, 3, 3, 4, 7, 7). Descendenții nodului 3 sunt:

- a. 5, 6, 7, 9, 10 b. 5, 6, 7 c. 6, 7 d. 4, 5, 6, 7, 9, 10

SUBIECTUL al II - lea

(40 de puncte)

1. Se consideră algoritmul alăturat, 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 valoarea afișată dacă se citește, în această ordine, numerele 7, 55, 72, 31, 17, 100, 93, 2. (6p.)

b) Dacă pentru `n` se citește valoarea 3, scrieți un set de numere distincte din intervalul $(0, 10^2]$ care pot fi citite în continuare, astfel încât, în urma executării algoritmului, să se afișeze valoarea 0. (6p.)

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

d) Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura `pentru...execută` cu o structură repetitivă cu test final. (6p.)

```
citește n (număr natural nenul)
```

```
m ← 0
```

```
pentru i ← 1, n execută
```

```
    citește x (număr natural nenul)
```

```
    d ← 2
```

```
    cât timp d ≤ [x/d] și x%d ≠ 0 execută
```

```
        d ← d+1
```

```
    ■
```

```
    dacă d ≤ [x/d] și x > m atunci
```

```
        m ← x
```

```
    ■
```

```
scrie m
```

2. Variabila **r** memorează simultan următoarele date despre fiecare dintre cele 20 de rase de cai dintr-o fermă: un cod, reprezentând rasa, numărul de exemplare din rasa respectivă și vârstele fiecărui exemplar. În fermă sunt maximum 10 exemplare din fiecare rasă. Știind că expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul și numărul de exemplare din prima rasă, respectiv vârsta celui de al patrulea exemplar din această rasă, scrieți definiția unei structuri cu eticheta **rasa**, care permite memorarea datelor despre o rasă de cai, și declarați corespunzător variabila **r**.

`r[0].cod r[0].nrExemplare r[0].varsta[3]`

(6p.)

3. Variabilele **i** și **j** sunt de tip întreg, iar variabila **a** memorează un tablou bidimensional de caractere, cu 6 linii și 6 coloane, numerotate începând de la 0, având inițial toate elementele egale cu **%**. Fără a utiliza alte variabile decât cele anterior menționate, scrieți secvența de instrucțiuni C/C++ de mai jos, înlocuind punctele de suspensie cu instrucțiuni adecvate astfel încât, în urma executării secvenței obținute, variabila **a** să memoreze tabloul alăturat.

`for (i=0; i<6; i++)
 for (j=0; j<6; j++)`

.....

```
* # ? * # ?  
# ? * # ? *  
? * # ? * #  
* # ? * # ?  
# ? * # ? *  
? * # ? * #
```

(6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Un număr natural se numește **aranjat** dacă are cel puțin o cifră pară, cel puțin o cifră impară și toate cifrele sale pare respectiv impare sunt grupate (apar pe poziții alăturate în scrierea numărului). De exemplu 71324, 56, 4813 sunt numere aranjate, dar 135 sau 826 sau 1234 nu sunt numere aranjate.

Subprogramul **aranjat** primește prin intermediul parametrului **n** un număr natural ($n \in [10, 10^9]$) și returnează valoarea 1 dacă numărul primit este aranjat sau 0 în caz contrar. Scrieți definiția completă a subprogramului C/C++.

Exemplu: dacă $n=7152$ sau $n=66233$, subprogramul va returna valoarea 1, iar dacă $n=51234$ sau $n=1759$ subprogramul va returna valoarea 0.

(10p)

2. Un **cuvânt marcat** este un cuvânt care începe cu secvența **...**, fără spațiu între puncte și cuvânt (de exemplu: **...gand** este un cuvânt marcat). Un text de cel mult 200 de caractere este format din cuvinte scrise cu litere mici ale alfabetului englez, cuvinte marcate și numere naturale, separate prin unul sau mai multe spații. Textul conține cel puțin două cuvinte și în text pot exista mai multe cuvinte marcate alăturate.

Scrieți un program C/C++ care citește de la tastatură un text de tipul precizat și îl transformă în memorie astfel încât:

- toate secvențele de spații sunt înlocuite de un singur spațiu;
- fiecare număr natural este înlocuit de cuvântul **numar**;
- fiecare cuvânt marcat se înlocuiește cu cel obținut prin eliminarea celor trei puncte și transformarea literelor rămase în majuscule;
- celelalte cuvinte rămân nemodificate.

Programul afișează pe ecran textul obținut.

(10p)

Exemplu: pentru textul: **...lumina ...blanda 7 ...ganduri si 3 umbre**

se afișează pe ecran: **LUMINA BLANDA numar GANDURI si numar umbre**

3. Fișierul **bac.txt** conține pe prima linie un număr natural **n** ($1 \leq n \leq 10^3$), iar pe a doua linie un șir de **n** numere naturale din intervalul $[1, 10^6]$, separate prin câte un spațiu. Se cere să se afișeze pe ecran, dacă există, o valoare **k** ($1 \leq k \leq n$) pentru care suma primelor **k** numere este egală cu suma următoarelor numere, de pe cea de a doua linie a fișierului. Dacă nu există o astfel de valoare se va afișa mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

Exemplu:

dacă fișierul conține numerele:

10

1 2 3 4 5 6 7 8 9 11

atunci se afișează pe ecran valoarea 7,

iar dacă fișierul conține numerele:

5

1 2 3 4 5

atunci pe ecran se afișează **nu exista**.

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia.

(2p)

b. Scrieți programul C/C++ corespunzător algoritmului proiectat.

(8p)