

Simulare pentru Examenul de bacalaureat noiembrie 2025

Proba E. d)
Informatică

Simulare 1

Filiera teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică

Filiera 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)

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 care este valoarea maximă pe care o poate avea expresia de mai jos în care x este o variabilă de tip întreg.
 $2 * x \% 10 * 2 \% 10$
a. 0 b. 7 c. 8 d. 9
2. Pentru definiția următoare a subprogramului f , indicați ce se va afișa în urma apelului $f(146563)$?

```
int f(int n)
{
    if(n==0) return 0;
    if(n%2) return 1+f(n/10);
    else return f(n/10);
}
```


a. 0 b. 2 c. 3 d. 5
3. Utilizând metoda backtracking, se generează în ordine crescătoare toate numerele de trei cifre, folosind doar cifre din mulțimea $\{0, 1, 2, 3, 4, 5\}$, cu restricția că două cifre alăturate nu pot avea aceeași paritate. Primele patru numere generate sunt: 101, 103, 105 și 121. Care sunt al 7-lea și penultimul număr generat?
a. 141, 543 b. 125, 541 c. 141, 541 d. 301, 525
4. Identificați care din secvențele următoare reprezintă șirul gradelor nodurilor unui graf complet.
a. 1 2 3 4 b. 1 2 12 12 c. 5 5 5 5 5 d. 4 4 4 4 4
5. Un arbore cu nodurile numerotate de la 1 la 10, este memorat cu ajutorul vectorului de tați $t=(2, 5, 5, 7, 0, 2, 3, 7, 6, 6)$. Numărul de lanțuri elementare de lungime maximă care leagă două noduri ale arborelui este:
a. 2 b. 4 c. 5 d. 6

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 în urma executării algoritmului, dacă se citesc, în această ordine, numerele 6, 145, 39, 7, 60, 257, 820, 0. (6p.)
- b) Dacă primul număr citit este 3, scrieți un set de numere distincte 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 **repetă** ... **până când** cu o structură repetitivă cu test inițial. (6p.)

```
citește a (nr. natural, a<10)
q ← 0
repetă
    citește x (nr. natural)
    y ← x
    cât timp y ≠ 0 execută
        dacă y%10 > a atunci
            q ← q+1
            y ← 0
        altfel
            y ← [y/10]
    până când x=0
scrie q
```

2. Considerăm că pentru o notă obținută de un elev se cunosc: disciplina (șir de maxim 30 de caractere), valoarea notei (număr natural) și data (zi, luna și an – numere naturale) la care a fost obținută nota. Scrieți declararea C/C++ a unei structuri cu eticheta **Nota** și a unei variabile **N** capabilă să memoreze 100 de note, astfel încât construcțiile de mai jos să fie corecte sintactic și să rețină informațiile referitoare la prima notă memorată.

`N[0].disciplina` `N[0].valoare`
`N[0].data.zi` `N[0].data.luna` `N[0].data.an` (6p.)

3. Scrieți șirul care se va afișa în urma execuției următoarei secvențe de program: (6p.)

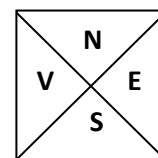
```
char s[30] = "simulareBACINFO", aux[30] = "";
strcpy(aux, strchr(s, 'B'));
s[8] = NULL;
strcpy(aux + 3, "-");
strcat(aux, s);
cout << aux;
```

SUBIECTUL al III-lea (30 de puncte)

1. Un număr natural n se numește număr **strict semiprim** dacă este produsul a exact două numere prime distincte. De exemplu, 6, 15, 10 sunt numere strict semiprime, iar 8, 12, 30 **nu** sunt numere strict semiprime. Subprogramul **strict_semiprim** are un singur parametru prin care primește un număr natural n ($2 \leq n \leq 10^6$) și returnează: 1, dacă n este un număr strict semiprim, sau 0, în caz contrar. Scrieți definiția completă a subprogramului C/C++.

(10p.)

2. Într-un tablou bidimensional cu n linii și n coloane, diagonala principală și diagonala secundară împart matricea în patru zone (nord, est, sud și vest), ca în imaginea alăturată. Elementele de pe diagonale nu aparțin nici unei zone. Scrieți un program C/C++ care citește de la tastatură numărul natural n ($n \in [3, 100]$) și $n \cdot n$ numere naturale din intervalul $[0, 10^9]$, elementele unui tablou bidimensional. Programul transformă tabloul în memorie, **interschimbând** elementele din **zona de est** cu cele din **zona de vest**, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, fiecare linie a tabloului pe câte o linie a ecranului, cu elementele fiecărei linii separate prin câte un spațiu.



Exemplu: dacă $n=5$ și tabloul este

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

se obține tabloul

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

(10p.)

3. Într-o lume tot mai digitalizată, compania tech *Cifralia Systems* dezvoltă o platformă inteligentă de analiză a datelor pentru orașele smart. Platforma primește zilnic milioane de coduri numerice care reprezintă identificatori pentru diverse entități urbane: contoare inteligente, autovehicule electrice, dispozitive etc. Pentru a face procesarea mai eficientă, codurile sunt grupate în funcție de **Indicele de Grupare** definit ca produsul dintre prima și ultima cifră a codului. Toate codurile cu același **Indice de Grupare** sunt puse în aceeași grupă.

Fișierul text **bac.in** conține cel mult un milion de numere naturale din intervalul $[10, 10^9]$ care reprezintă lista de coduri încărcată pe platformă. Se cere să se afișeze pe ecran, câte grupe diferite se pot forma, dacă aplicăm regula Indicelui de Grupare pe lista de coduri. Proiectați un algoritm eficient din punctul de vedere al timpului de executare.

Exemplu: dacă fișierul conține valorile următoare: 253 1343 217 341 6034 1034816 83 62951 44, pe ecran se afișează: 5

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