

Subiectul II (30 de puncte)

Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera care corespunde răspunsului corect.

1. Câte dintre vârfurile grafului neorientat G , reprezentat prin matricea de adiacență alăturată, au gradul un număr par? **(4p.)**
- | | | | | |
|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
- a. 3 b. 1 c. 2 d. 5
2. Într-o stivă inițial vidă au fost executate următoarele operații:
`push 3; push 7; pop; push 5; push 1;`
unde `push a` reprezintă operația prin care valoarea `a` se adaugă în stivă, iar `pop` reprezintă operația prin care se extrage un element din stivă.
Care este elementul situat în vârful stivei? **(4p.)**
- a. 1 b. 5 c. 7 d. 3

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

3. Pentru reprezentarea unui arbore cu rădăcină cu 10 noduri, etichetate cu numere naturale de la 1 la 10, se utilizează vectorul de tați: $TATA = (4, 8, 8, 0, 10, 4, 8, 6, 2, 6)$. Care sunt frunzele arborelui? **(6p.)**
4. Ce se afișează pe ecran în urma executării secvenței de program alăturate știind că variabila `i` este de tip `char`? **(6p.)**
- ```
for (i='a'; i<='z'; i++)
 if (i<'d')
 cout<<i; | printf("%c",i);
```
5. Scrieți un program **C/C++** care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 23$ ) și apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , astfel încât fiecare element situat pe o linie  $i$  ( $1 \leq i \leq n$ ) și pe o coloană  $j$  ( $1 \leq j \leq n$ ) va fi egal cu suma dintre  $i$  și  $j$ . Programul va afișa matricea pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu. **Exemplu:** dacă  $n=4$ , se va afișa matricea alăturată. **(10p.)**
- |   |   |   |   |
|---|---|---|---|
| 2 | 3 | 4 | 5 |
| 3 | 4 | 5 | 6 |
| 4 | 5 | 6 | 7 |
| 5 | 6 | 7 | 8 |

**Subiectul III (30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Cea mai mare dintre valorile strict negative memorate într-un tablou unidimensional NU există dacă: **(4p.)**
  - a. În tablou sunt memorate numai valori strict negative
  - b. În tablou există cel puțin o valoare strict negativă
  - c. În tablou sunt memorate valori nenule pozitive și negative
  - d. În tablou sunt memorate numai valori strict pozitive

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Știind că  $p$  este un vector cu 3 componente întregi (vector declarat global) cu ce trebuie înlocuite  $\alpha$  și  $\beta$  în definiția subprogramului  $G$  astfel încât în urma apelului  $G(0)$  să se afișeze toate numerele de 3 cifre, fiecare cifră fiind nenulă. Fiecare număr va fi afișat o singură dată. **(6p.)**

```
void G(int k)
{
 int i;
 for(i=1; i<= α ; i++)
 {
 p[k]=i;
 if(β) G(k+1);
 else
 cout<<p[0]<<p[1]<<p[2]<<endl; |
 printf("%d%d%d\n", p[0], p[1], p[2]);
 }
}
```
3. Scrieți un subprogram **MULT**, cu doi parametri, care primește prin intermediul parametrului  $a$  un tablou unidimensional cu cel mult 100 de elemente, numere naturale de cel mult 4 cifre fiecare, și prin intermediul parametrului  $n$  un număr natural nenul,  $n < 100$ , ce reprezintă numărul de elemente din tablou. Subprogramul verifică dacă tabloul conține toate valorile din mulțimea  $\{1, 2, \dots, n\}$ . În caz afirmativ subprogramul returnează valoarea 1, iar altfel subprogramul returnează valoarea 0. **(10p.)**
4. Fișierul text **numere.txt** conține pe prima linie un număr natural  $n$  ( $0 < n < 100000$ ), iar pe a doua linie un șir de  $n$  numere naturale de o singură cifră, separate prin câte un spațiu. Șirul conține cel puțin o valoare pară și una impară.
  - a) Scrieți un program C/C++ care determină în mod eficient, din punct de vedere al timpului de executare, cea mai mare cifră pară și cea mai mică cifră impară dintre cele situate pe a doua linie a fișierului. Cifrele determinate vor fi afișate pe ecran, pe o singură linie, separate printr-un spațiu.  
**Exemplu:** dacă fișierul **numere.txt** are următorul conținut:  
7  
3 5 2 1 6 3 1  
atunci pe ecran se vor afișa valorile: 6 1. **(6p.)**
  - b) Descrieți succint, în limbaj natural, metoda de rezolvare folosită, explicând în ce constă eficiența ei (3 – 4 rânduri). **(4p.)**