

Laborator 9

1. Fie $G = (V, E)$ un graf orientat reprezentat prin matrice de adiacenta:

```
typedef struct graf {  
    int n;  
    int **mat;  
} graf;
```

Sa se implementeze urmatoarele operatii pe acesta:

- `graf *init(int n)` → initializeaza graful
- `void add_edge(graf *g, int i, int j)` → adauga o muchie intre nodurile i si j
- `void del_edge(graf *g, int i, int j)` → elimina muchia dintre nodurile i si j
- `int in_degree(graf *g, int i)` → intoarce gradul interior al nodului i
- `int ext_degree(graf *g, int i)` → intoarce gradul exterior al nodului i

2. Sa se implementeze algoritmul Floyd-Warshall pentru determinarea drumului de cost minim intre oricare doua noduri ale unui graf orientat.

```
FLOYD-WARSHALL'(W)  
1   $n = W.rows$   
2   $D = W$   
3  for  $k = 1$  to  $n$   
4      for  $i = 1$  to  $n$   
5          for  $j = 1$  to  $n$   
6               $d_{ij} = \min(d_{ij}, d_{ik} + d_{kj})$   
7  return  $D$ 
```

3. Fiind dat un graf orientat ponderat, sa se scrie o functie care calculeaza lungimea drumul minim intre doua varfuri ale acestuia. De asemenea, sa se determine si calea intre aceste doua noduri.

4. Într-o țară sunt n orașe, numerotate de la 1 la n , unite între ele prin m șosele bidirecționale de lungimi cunoscute, între oricare două orașe existând drum, fie șosea directă, fie prin alte orașe. O firmă dorește să-și stabilească sediul în unul dintre orașe, astfel încât suma lungimilor drumurilor minime de la orașul în care se află sediul la toate celelalte orașe să fie minimă. Determinați orașul care va fi ales pentru sediul firmei. Dacă sunt mai multe orașe care pot fi alese, se va alege cel cu numărul de ordine mai mic.

IN:

6 9

1 3 5

1 4 5

1 5 4

2 3 5

2 4 1

2 5 2

3 6 2

4 6 4

5 6 4

OUT:

2