

***МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА
ПОЛІТЕХНІКА”***

Кафедра систем штучного інтелекту

***Розрахункова робота
з дисципліни
«Дискретна математика»***

Виконав:
студент групи КН-114
Мороз Павло

Викладач:
Мельникова Н.І

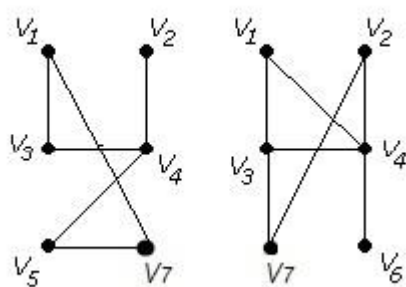
Львів – 2019 р.

ІНДИВІДУАЛЬНІ ЗАВДАННЯ

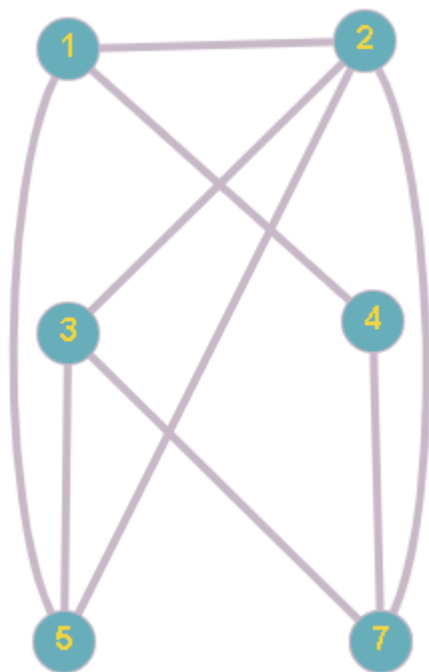
Завдання № 1

Виконати наступні операції над графами: 1) знайти доповнення до першого графу, 2) об'єднання графів, 3) кільцеву сумму $G1$ та $G2$ ($G1+G2$), 4) розмножити вершину у другому графі, 5) виділити підграф A - що складається з 3-х вершин в $G1$ 6) добуток графів.

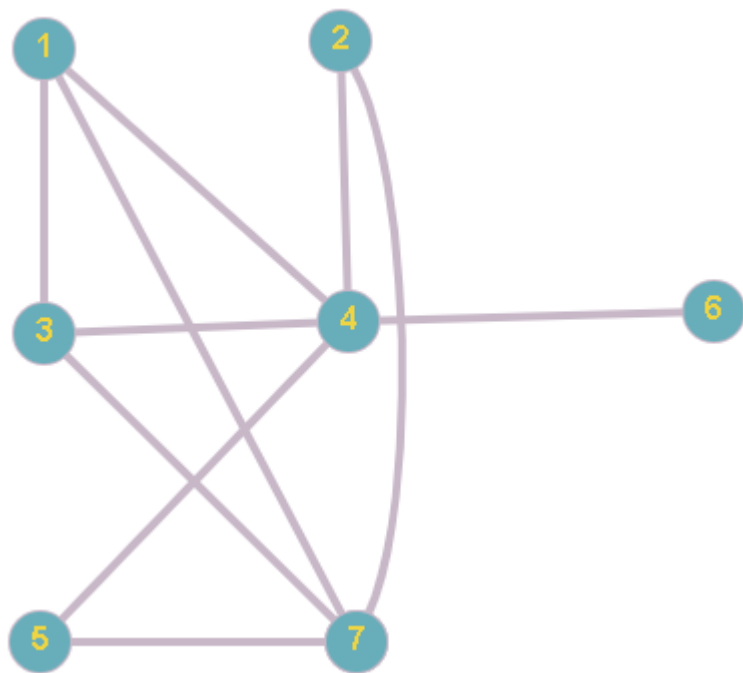
17)



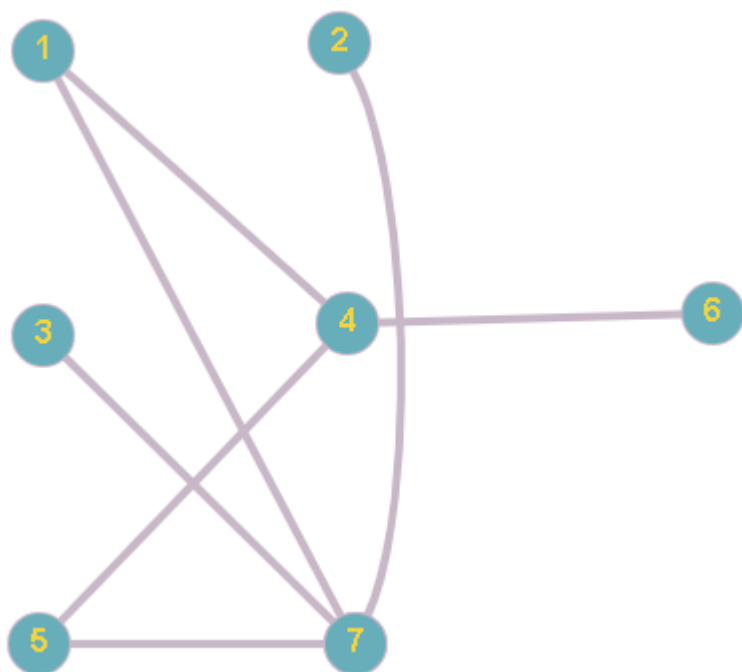
1.1)



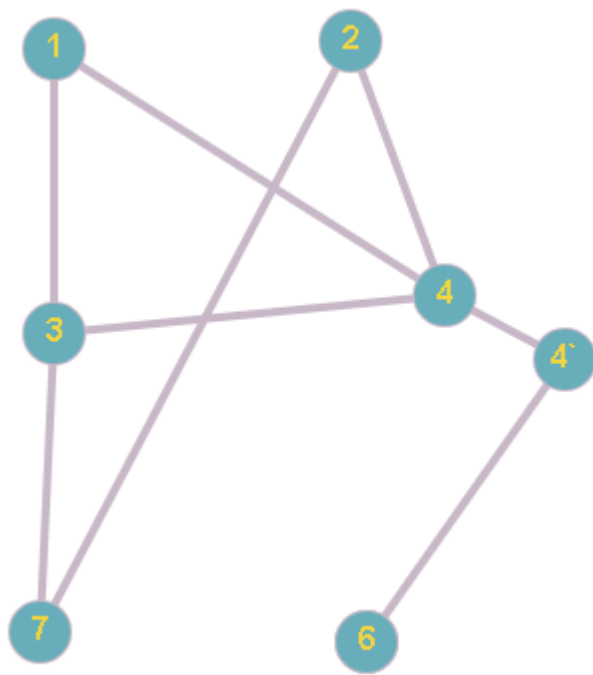
1.2)



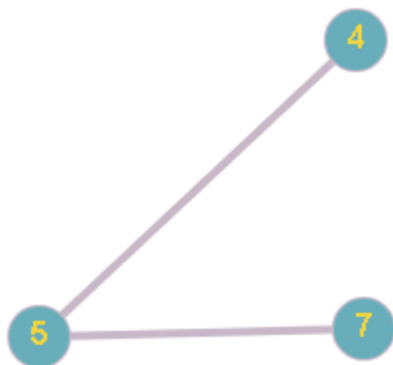
1.3)



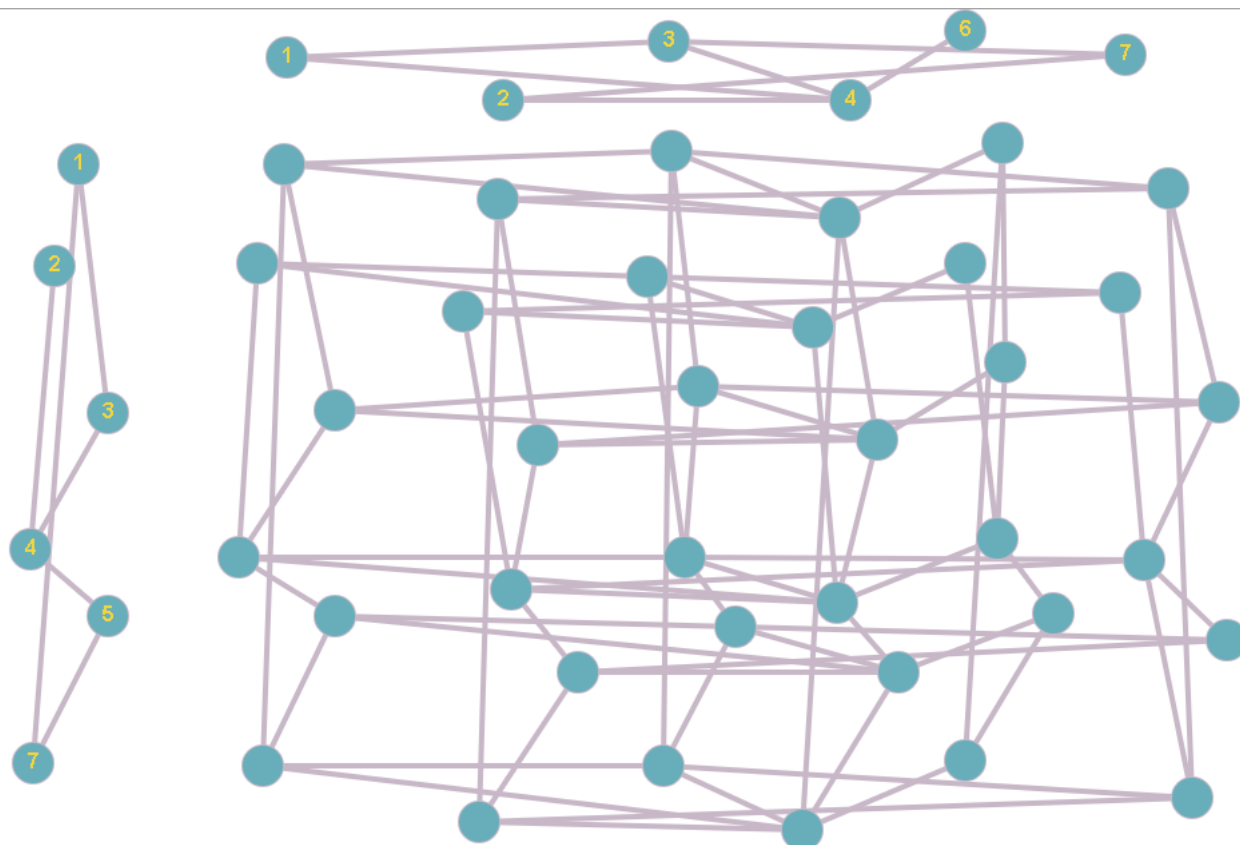
1.4)



1.5)



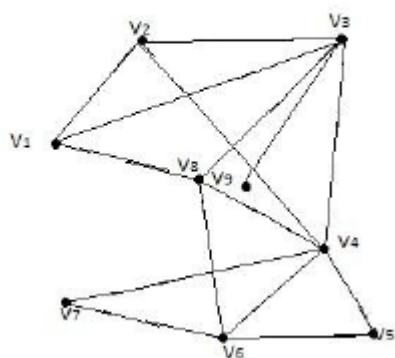
1.6)



Завдання № 2

Скласти таблицю суміжності для орграфа.

17)



	V1	V2	V3	V4	V5	V6	V7	V8	V9
V1	0	1	1	0	0	0	0	1	0
V2	1	0	1	1	0	0	0	0	0
V3	1	1	0	1	0	0	0	1	1
V4	0	1	1	0	1	1	1	1	0
V5	0	0	0	1	0	1	0	0	0
V6	0	0	0	1	1	0	1	1	0
V7	0	0	0	1	0	1	0	0	0
V8	1	0	1	1	0	1	0	0	0
V9	0	0	1	0	0	0	0	0	0

Завдання № 3

Для графа з другого завдання знайти діаметр.

Діаметр графа:3, бо найбільша відстань між вершинами V1 та V5.

Завдання № 4

Для графа з другого завдання виконати обхід дерева вглиб (варіант закінчується на непарне число) або вшир (закінчується на парне число).

Вершина	Номер	Стек
1	0	1
2	1	1,2
3	2	1,2,3
4	3	1,2,3,4
5	4	1,2,3,4,5
6	5	1,2,3,4,5,6
7	6	1,2,3,4,5,6,7
—	—	1,2,3,4,5,6
8	7	1,2,3,4,5,6,8
—	—	1,2,3,4,5,6
—	—	1,2,3,4,5
—	—	1,2,3,4
—	—	1,2,3
9	8	1,2,3,9
—	—	1,2,3
—	—	1,2
—	—	1
—	—	—

Код програми

```

#include <bits/stdc++.h>
using namespace std;
map <int, set<int> > m;
queue <int> q;
set <int> s;
int n,a,b;
int main(){
    cout<<"Enter number of tops\n";
    cin>>n;
    cout<<"Enter start and end of edge\n";
    for(int i=1;i<=n;i++){
        cin>>a>>b;
        m[a].insert(b);
    }
    int start;
    cout<<"Enter top to dfs\n";
    cin>>start;
    q.push(start);
    while(!q.empty()){
        int t = q.front();
        q.pop();
        s.insert(t);
        for(auto it = m[t].begin();it!=m[t].end();++it){
            if(s.count(*it)==0){q.push(*it);}
        }
    }
    cout<<"Tops linked with "<<start<<" top: ";
    for(auto it = s.begin();it!=s.end();++it)
        cout<<*it<<' ';
}

```

Результат роботи коду

```

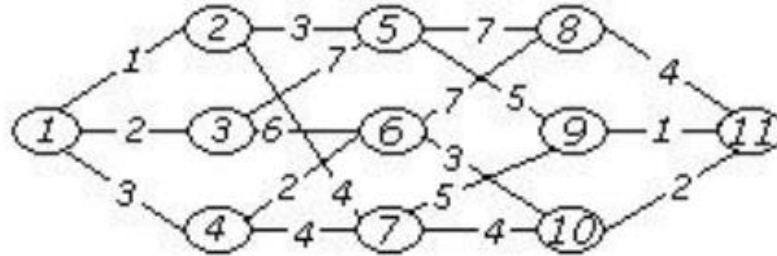
Enter number of tops and edges
9 15
Enter start and end of edge
1 2
1 3
1 8
2 3
2 4
3 4
3 8
3 9
4 5
4 6
4 7
4 8
5 6
6 7
6 8
Enter top to dfs
1
Tops linked with 1 top: 1 2 3 4 5 6 7 8 9
Process returned 0 (0x0)   execution time : 99.306 s
Press any key to continue.

```

Завдання № 5

Знайти двома методами (Краскала і Прима) мінімальне остове дерево графа.

17)



Прима:

$V = \{1, 2, 3, 4, 6, 5, 10, 11, 9, 8, 7\}$, $E = \{\{1, 2\}, \{1, 3\}, \{1, 4\}, \{4, 6\}, \{2, 5\}, \{6, 10\}, \{10, 11\}, \{11, 9\}, \{11, 8\}, \{4, 7\}\}$. Minimalost=25;

Код програми

```
using namespace std;
int ways[14], a[14][14], vis[14] = {0}, tops[14] = {0}, topNumb = 0, val, s, f;
int min_array(int n) {
    int min = INT_MAX, nmin = 0;
    for(int i=1; i<=n; i++) {
        if(ways[i] < min && vis[i] == 0) {
            min = ways[i]; nmin = i;
        }
    }
    return nmin;
}
int main() {
    int n, m;

    cout<<"Enter amount of tops and edges of graph\n";
    cin>>n>>m;
    cout<<"Enter number of start top, end top and edge weight between them\n";
    for(int i=0; i<=n; i++)
        ways[i] = INT_MAX;
    for(int i = 1; i<=m; i++) {
        cin>>s>>f>>val;
        a[s][f] = val;
        a[f][s] = val;
    }
    ways[1] = 0;
    int current = 1;
    while(current != 0) {
        vis[current] = 1;
        for(int i=1; i<=n; i++) {
            if(a[current][i] != 0 && vis[i] == 0) ways[i] = min(ways[i], a[current][i]);
        }
        topNumb++;
        tops[topNumb] = current;
        current = min_array(n);
    }
    cout<<"The order of tops after algorithm passing\n";
    int sum = 0;
    for(int i=1; i<=n; i++) {
        sum += ways[i];
        cout<<tops[i]<<" -> ";
    }
    cout<<tops[n];
    cout<<"\nLength of minimal frame: "<<sum + ways[n];
}
```

Результат роботи коду

Enter amount of tops and edges of graph
 11 18
 Enter number of start top,end top and edge weight between them
 1 2 1
 1 3 2
 1 4 3
 2 5 3
 3 5 7
 3 6 6
 4 6 2
 4 7 4
 2 7 4
 5 8 7
 5 9 5
 6 8 7
 6 10 3
 7 9 5
 7 10 4
 8 11 4
 9 11 1
 10 11 2
 The order of tops after algorithm passing
 1 -> 2 -> 3 -> 4 -> 6 -> 5 -> 10 -> 11 -> 9 -> 7 -> 8
 Length of minimal frame: 25

Краскала:

```
#include <bits/stdc++.h>
using namespace std;

int main(){
    int n, m;
    cin >> n >> m;
    vector <vector <int> > edges(m, vector<int>(3));
    for (int i = 0; i < m; ++i)
        cin >> edges[i][1] >> edges[i][2] >> edges[i][0];
    sort(edges.begin(), edges.end());
    vector <int> comp(n);
    vector <pair<int,int>> answer;
    for (int i = 1; i <= n; ++i)
        comp[i] = i;
    int ans = 0;
    for (auto & edge: edges)
    {
        int weight = edge[0];
        int start = edge[1];
        int end = edge[2];
        if (comp[start] != comp[end])
        {
            ans += weight;
            int a = comp[start];
            int b = comp[end];
            answer.push_back(make_pair(a,b));
            for (int i = 1; i <= n; ++i)
                if (comp[i] == b)
                    comp[i] = a;
        }
    }
    for(int i=0;i<answer.size();i++)
        cout<<answer[i].first<<' '<<answer[i].second<<endl;
    cout<<"Ans "<<ans;
}
```

Результат роботи

11 18
 1 2 1
 1 3 2
 1 4 3
 2 5 3
 3 5 7
 3 6 6
 4 6 2
 4 7 4
 2 7 4
 5 8 7
 5 9 5
 6 8 7
 6 10 3
 7 9 5
 7 10 4
 8 11 4
 9 11 1
 10 11 2
 1 2
 9 11
 1 3
 4 6
 10 9
 1 4
 1 5
 1 10
 1 7
 8 1
 Ans 25

$E = \{\{1,2\}, \{9,11\}, \{1,3\}, \{4,6\}, \{10,11\}, \{1,4\}, \{2,5\}, \{6,10\}, \{4,7\}, \{11,8\}\}$, $V = \{1,2,9,11,3,4,6,10,4,5,6,7,8\}$; Minimalost=25;

Завдання № 6

Розв'язати задачу комівояжера для повного 8-ми вершинного графа методом «іди у найближчий», матриця вагів якого має вигляд:

17)

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

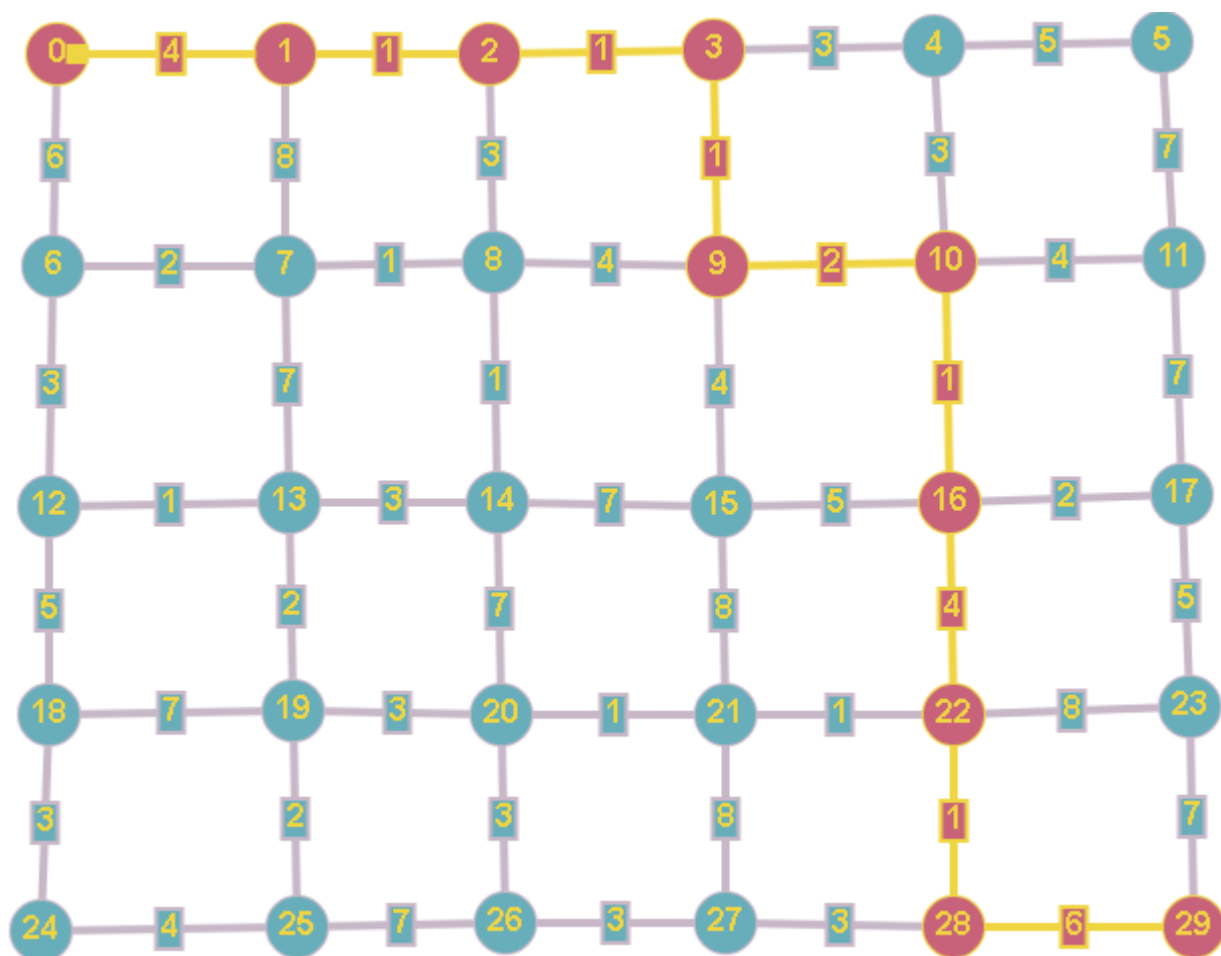
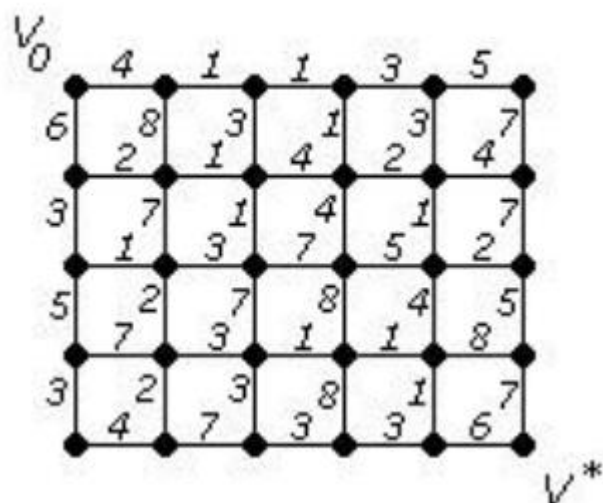
Найменший цикл з вагою 19, який має таку послідовність

3 8 4 7 6 1 5 2 3.

Завдання № 7

За допомогою алгоритму Дейкстри знайти найкоротший шлях у графі між парою вершин V_0 і V^* .

17)



Мінімальна відстань:21

Код до алгоритму Дейкстри

```
#include <iostream>
#include <cstdio>
using namespace std;
int n,i,j,a[101][101],b[101],c[101]={0},x,y,from[102]={0},m,w,s,f;
void dejkstra(int v)
{
    c[v]=1;
    for(i=0;i<n;i++)
        if(a[v][i]!=-1 && c[i]==0 && v!=i)
        {
            if(b[v]+a[v][i]<b[i]){
                b[i]=min(b[i],b[v]+a[v][i]);from[i] = v;}
        }
    int min_ar()
    {
        int min=1000000,nm=-1;
        for(i=0;i<n;i++)
            if(min>b[i] && c[i]==0){min=b[i];nm=i;}
        return nm;
    }
}

int out(int v){
    return from[v];
}

int main()
{
    cout<<"Enter number of tops and edges, start and finish tops\n";

    cin>>n>>m>>x>>y;
    cout<<"Enter list of edges\n";
    cout<<n;
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            a[i][j] = -1;
            b[i] = INT_MAX;
        }
    }
    for(int i=1;i<=m;i++){
        cin>>s>>f>>w;
        a[s][f] = w;
        a[f][s] = w;
    }
    if(a[x][y]==0){cout<<0;return 0;}

    b[x]=0;
    int t=1;
    while(t==1)
    {
        t=0;
        int v=min_ar();
        if(c[v]==0 && v>=0 ) {dejkstra(v); t=1;}
    }

    cout<<"The shortest way: "<<b[y]<<endl;
    cout<<"Tops on the shortest way: ";
    int curr = y;
    int path[n*n] = {0},sz = 0;
    while(curr!=x){
        path[sz] = curr;
        sz++;
        curr = out(curr);
    }
    for(int i=sz-1;i>=0;i--)
        cout<<path[i]<<' ';
}
```

Результат роботи

8 14 1
9 10 2
9 15 4
10 11 4

10 16 1
11 17 7
12 13 1
12 18 5
13 14 3
13 19 2
14 15 7
14 20 7
15 16 5

15 21 8
16 17 2
16 22 4
17 23 5
18 19 7
18 20 3
19 20 3
19 25 2
20 21 1
20 26 3

21 22 1
21 27 8
22 28 1
22 23 8
23 29 7
24 25 4
25 26 7
26 27 3
27 28 3
28 29 6

The shortest way: 21

Tops on the shortest way: 1 2 3 9 10 16 22 28 29

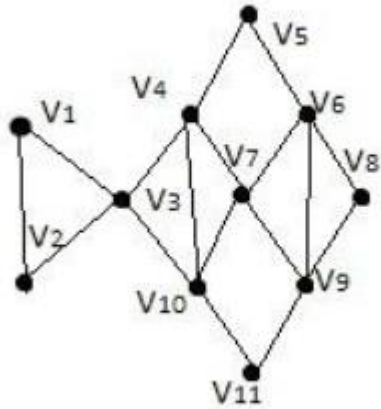
Process returned 0 (0x0) execution time : 11.742 s

Press any key to continue.

Завдання № 8

Знайти ейлеровий цикл в ейлеровому графі двома методами: а) Флері; б) елементарних циклів.

17)



a) 1-3-10-11-9-8-7-10-4-7-6-5-4-3-2-1

б) 1-2, 2-3, 3-1, 3-4, 4-5, 5-6, 6-7, 7-4, 4-10, 10-3, 6-8, 8-9, 9-7, 9-11, 11-10

Код до алгоритму Флері


```

#include <bits/stdc++.h>
using namespace std;
int a[107][107],m,n,s,f;
stack <int> st;
vector <int> res;

int main(){
    cin>>n>>m;
    for(int i=1;i<=m;i++){
        cin>>s>>f;
        a[s][f] = 1;
        a[f][s] = 1;
    }
    st.push (1);
    while (!st.empty()) {
        int v = st.top();
        int i;
        for (i=1; i<=n; ++i)
            if (a[v][i])
                break;
        if (i == n+1) {
            res.push_back (v);
            st.pop();
        }
        else {
            --a[v][i];
            --a[i][v];
            st.push (i);
        }
    }
    for(int i=0;i<res.size();i++){
        cout<<res[i]<<' ';
    }
}

```

Результат роботи

 C:\Users\Admin\Desktop\Flery\bin\Debug\Flery.exe

11 17

1 2

1 3

2 3

3 4

3 10

4 10

4 5

4 7

10 7

10 11

5 6

6 7

7 9

9 11

6 9

6 8

9 8

1 3 10 11 9 8 6 9 7 10 4 7 6 5 4 3 2 1

Завдання №9

Спростити формулу $x\bar{y}\vee\overline{x\bar{y}}\vee yz$

$$f = x\bar{y}z \vee x\bar{y}\bar{z} \vee \bar{x}y\bar{z} \vee \bar{x}\bar{y}\bar{z} \vee xyz \vee \bar{x}yz$$

$$\bar{x}\bar{y}\bar{z} \qquad \bar{y}\bar{z}$$

$$x\bar{y}\bar{z} \qquad \bar{x}\bar{z}$$

$$\bar{x}y\bar{z} \qquad x\bar{y}$$

$$x\bar{y}z \qquad \bar{x}y$$

$$\bar{x}yz \qquad xz$$

$$\overline{x\bar{y}z} \qquad yz$$

$$f = \bar{y}\bar{z} \vee \bar{x}\bar{z} \vee x\bar{y} \vee \bar{x}y \vee xz \vee yz$$