

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

Кафедра систем штучного інтелекту

Лабораторна робота №4

з дисципліни «Дискретна математика»

Виконала:

студент групи КН-114

Серкіз Людмила

Викладач:

Мельникова Н.І.

Львів – 2019 р.

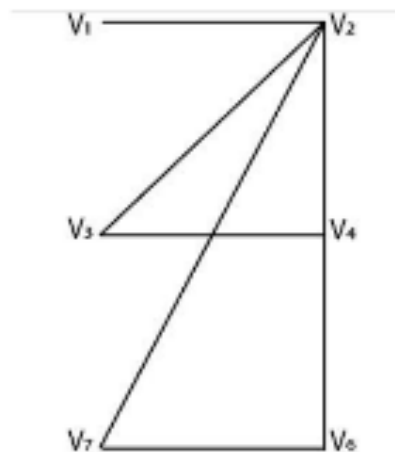
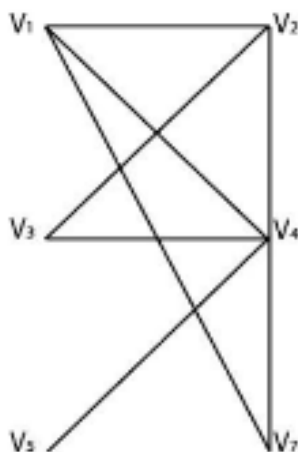
Тема: Основні операції над графами. Знаходження остова мінімальної ваги за алгоритмом Пріма-Краскала

Мета роботи: набуття практичних вмінь та навичок з використання алгоритмів Пріма і Краскала.

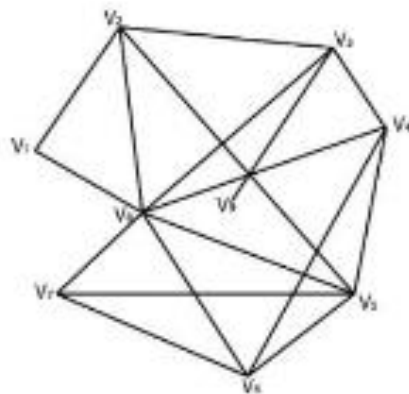
Завдання № 1. Розв'язати на графах наступні задачі:

1. Виконати наступні операції над графами:

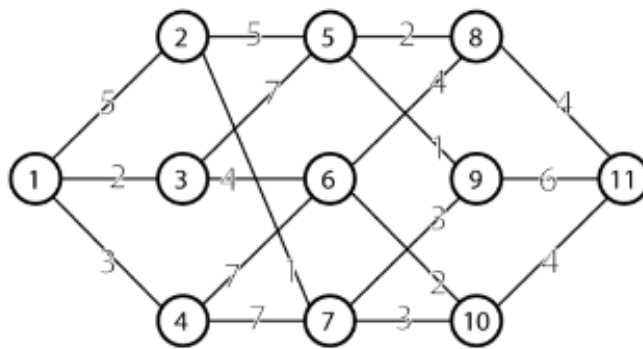
- 1) знайти доповнення до першого графу,
 - 2) об'єднання графів,
 - 3) кільцеву суму $G1$ та $G2$ ($G1+G2$),
 - 4) розщепити вершину у другому графі,
 - 5) виділити підграф A , що складається з 3-х вершин в $G1$ і знайти стягнення A в $G1$ ($G1 \setminus A$), 6) добуток графів.
-



2. . Знайти таблицю суміжності та діаметр графа.

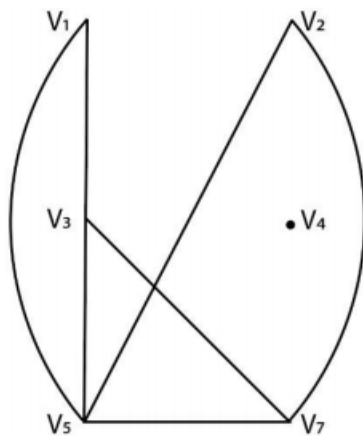


3. . Знайти двома методами (Краскала і Прима) мінімальне остове дерево графа.



Розв'язування:

1)Доповнення до першого графу (G_3):

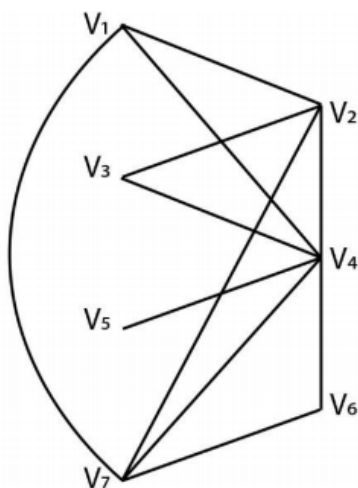


$$X_3 = \{V_1, V_2, V_3, V_4, V_5, V_7\}$$

$$W_3 = \{(V_1, V_3), (V_1, V_5), (V_2, V_5), (V_2, V_7),$$

$$(V_3, V_5), (V_3, V_7), (V_5, V_7)\}$$

2)Об'єднання графів G_1 і G_2 ($G_1 \cup G_2$):



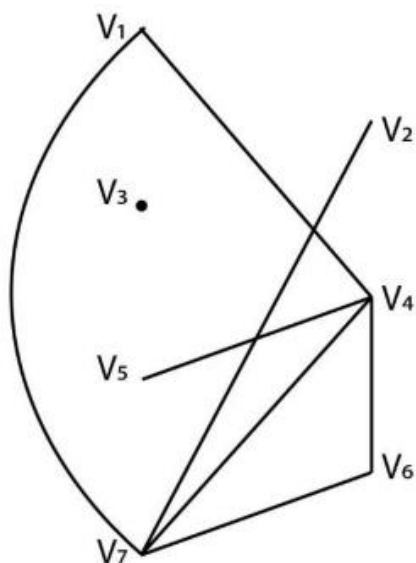
$$X_4 = \{V_1, V_2, V_3, V_4, V_5, V_6, V_7\}$$

$$W_4 = \{(V_1, V_2), (V_1, V_4), (V_1, V_7), (V_2, V_3),$$

$$(V_2, V_4), (V_2, V_7), (V_3, V_4), (V_4, V_5), (V_4, V_6),$$

$$(V_4, V_7), (V_6, V_7)\}$$

3) Кільцева сума G_1 і G_2 ($G_1 \oplus G_2$):



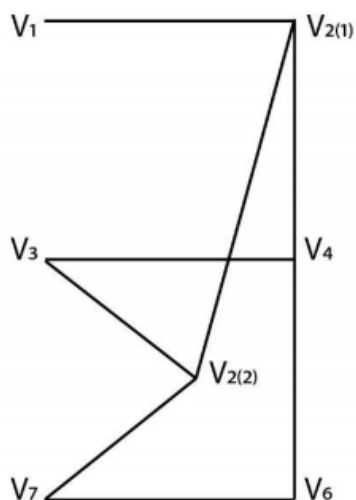
$$X_5 = \{V_1, V_2, V_3, V_4, V_5, V_6, V_7\}$$

$$W_5 = \{(V_1, V_4), (V_1, V_7),$$

$$(V_2, V_7), (V_4, V_5), (V_4, V_6),$$

$$(V_4, V_7), (V_6, V_7)\}$$

4) Розчепимо вершину V_2 на $V_{2(1)}$ і $V_{2(2)}$:



$$X_6 = \{V_1, V_{2(1)}, V_{2(2)}, V_3, V_4, V_6, V_7\}$$

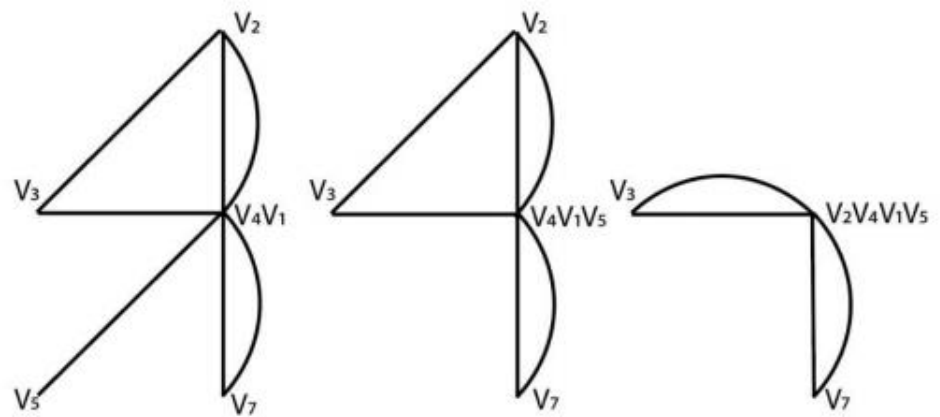
$$W_6 = \{(V_1, V_{2(1)}), (V_{2(1)}, V_{2(2)}), (V_{2(2)}, V_3), (V_{2(1)}, V_4),$$

$$, (V_{2(2)}, V_7), (V_3, V_4), (V_4, V_6), (V_6, V_7)\}$$

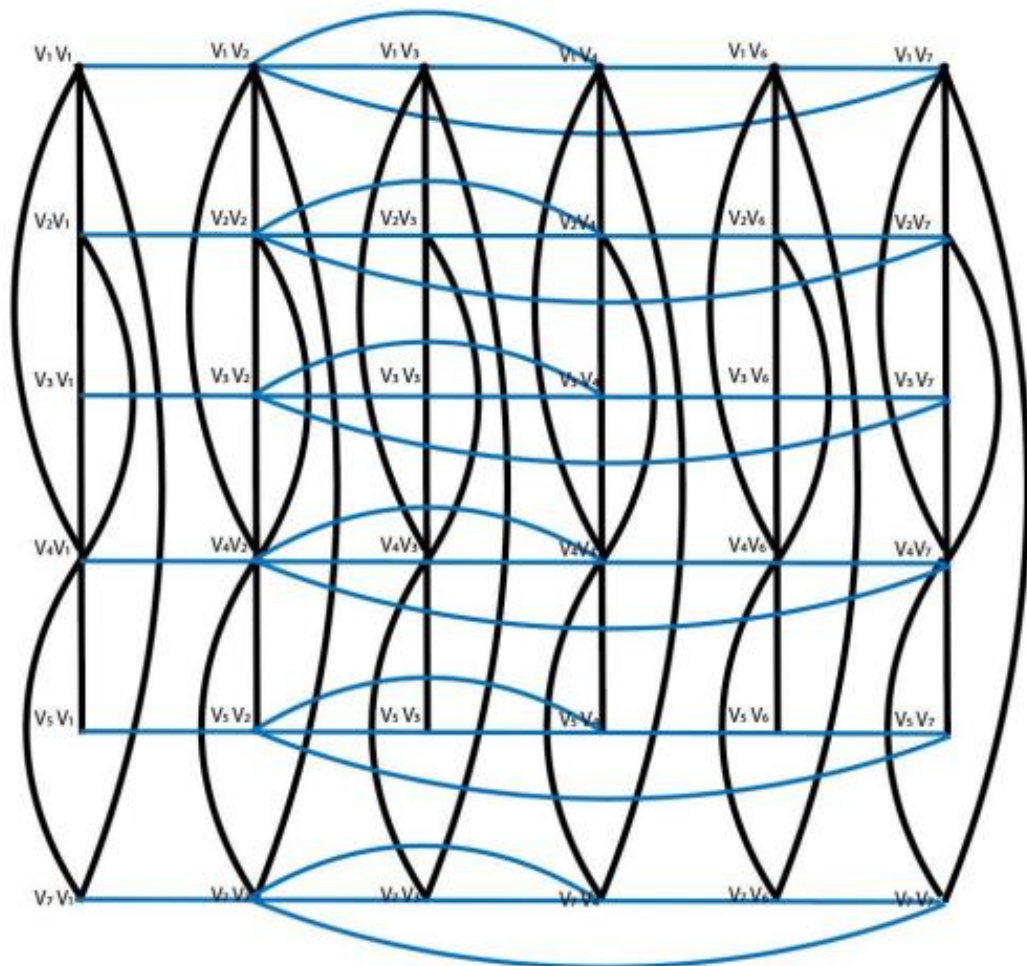
5) Підграф А



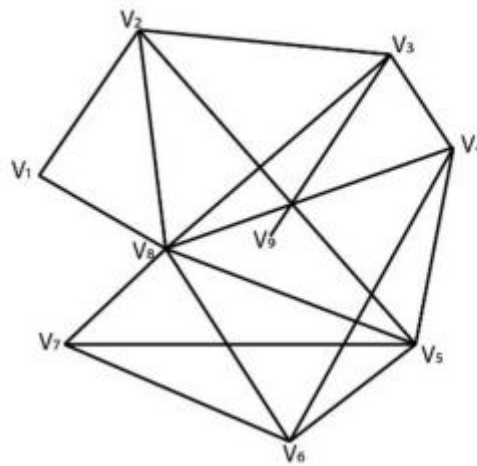
Виконаємо стягнення цього підграфа з графа G_1 ($G_1 \setminus A$):



б) Множення графів G_1 і G_2 ($G_1 \times G_2$):



2.



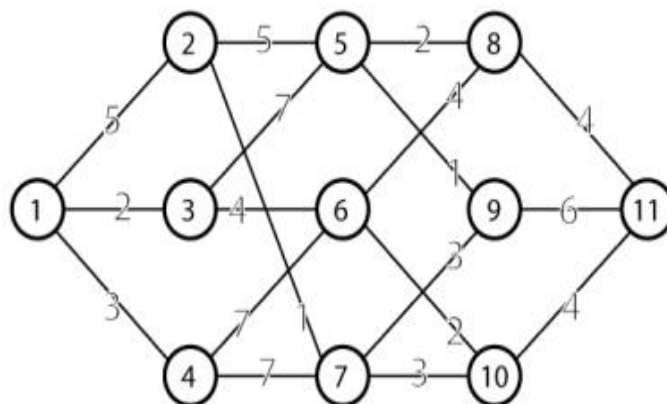
Таблиця інцидентності :

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9
V_1	0	1	0	0	0	0	0	1	0
V_2	1	0	1	0	1	0	0	1	0
V_3	0	1	0	1	0	0	0	1	1
V_4	0	0	1	0	1	1	0	1	0
V_5	0	1	0	1	0	1	1	1	0
V_6	0	0	0	1	1	0	1	1	0
V_7	0	0	0	0	1	1	0	1	0
V_8	1	1	1	1	1	1	1	0	0
V_9	0	0	1	0	0	0	0	0	0

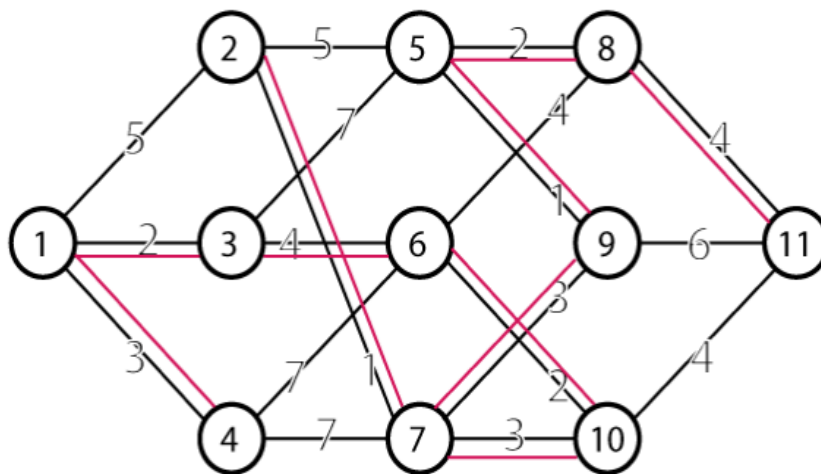
	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9
V_1	-	1	2	2	2	2	2	1	3
V_2	1	-	1	2	1	2	2	1	2
V_3	2	1	-	1	2	2	2	1	1
V_4	2	2	1	-	1	1	2	1	2
V_5	2	1	2	1	-	1	1	1	3
V_6	2	2	2	1	1	-	1	1	3
V_7	2	2	2	2	1	1	-	1	3
V_8	1	1	1	1	1	1	1	-	2
V_9	3	2	1	2	3	3	3	2	-

Діаметр даного графа : 3.

3.

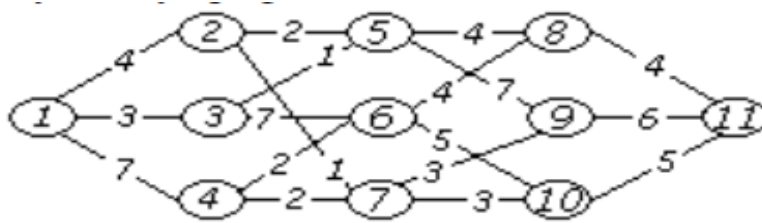


Вага остового дерева: $2+3+4+2+3+3+1+1+2+4=25$



Додаток 2

За алгоритмом Прима знайти мінімальне остове дерево графа. Етапи розв'язання задачі виводити на екран. Протестувати розроблену програму на наступному графі:



```

1  #include<iostream>
2  using namespace std;
3  int main()
4  { int n,n0;
5    cout<<"Enter the number of vertices: ";
6    cin>>n;
7    cout<<"\nEnter the number of edges: " << endl;
8    cin>>n0;
9    cout<<"\nnow\nenter the data in order: \nA(1) A(2) A(1)A(2)(weight of edges)\n";
10   int ** arr;
11   arr=new int*[n] ;
12   for(int i=0;i<n;i++){
13     arr[i]=new int[n];
14     for(int j=0;j<n;j++){
15       { arr[i][j]=0;}
16     }
17     int a1,a2,w;
18     for(int i=0;i<n0;i++){
19       cin>> a1 >> a2 >> w;
20       arr[a1-1][a2-1]=w;
21       arr[a2-1][a1-1]=w;
22     }
23     int* mas;
24     mas=new int[n];
25     for(int i=0;i<n;i++){
26       mas[i]=0;}
27     mas[0]=1;
28     int sum=0;

```

```

29 cout<<"\n\n";
30 for(int r=0,t=0;r!=(n-1);)
31 { t=0;
32   for(int i=0;i<n;i++){
33     if(mas[i]!=0){
34       for(int j=0;j<n;j++){
35         if(arr[i][j]!=0){
36           if(t==0){
37             a1=i;
38             a2=j;
39             w=arr[i][j];
40             t++;
41           }
42           else{
43             if(arr[i][j]<w){
44               a1=i;
45               a2=j;
46               w=arr[i][j];
47             }
48           }
49         }
50       }
51     }
52   }
53   sum+=w;
54   cout<<"\nA("<<a1+1<<")->A("<<a2+1<<")="<<w;
55   mas[a2]=a2+1;
56   arr[a1][a2]=0;

```

```

57   arr[a2][a1]=0;
58   r=0;
59   for(int q=0;q<n;q++){
60     if(mas[q]!=0){
61       r++;}
62   }
63 }
64 for(int i=0;i<n;i++){
65   if(mas[i]==0){a2=i;}
66 } for(int i=0,t=0;i<n;i++){
67   if(arr[i][a2]!=0){
68     if(t==0){
69       a1=i;
70       w=arr[i][a2];
71       t++;
72     }
73     else{
74       if(arr[i][a2]<w){
75         a1=i;
76         w=arr[i][a2];
77       }
78     }
79   }
80 }
81 sum+=w;
82 cout<<"\nA("<<a1+1<<")->A("<<a2+1<<")="<<w;
83 cout<<"\n\nminimal weight:"<<sum;
84 cout<<"\n\n\n";

```



```
Enter the number of vertices: 11

Enter the number of edges:
18

now
enter the data in order:
A(1) A(2) A(1)A(2) (weight of edges)
1 2 4
1 3 3
1 4 7
2 5 2
2 7 1
3 5 1
3 6 7
4 6 2
4 7 2
5 8 4
5 9 7
6 8 4
6 10 5
7 9 3
7 10 3
8 11 4
9 11 6
10 11 5
```

```
A(1)->A(3)=3
A(3)->A(5)=1
A(5)->A(2)=2
A(2)->A(7)=1
A(7)->A(4)=2
A(4)->A(6)=2
A(7)->A(9)=3
A(7)->A(10)=3
A(1)->A(2)=4
A(5)->A(8)=4
A(8)->A(11)=4
```

```
minimal weight:29
```

```
...Program finished with exit code 0
Press ENTER to exit console.
```

Висновок: Я практикувала і отримала навички використання алгоритмів Прима і Краскала