

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

Кафедра систем штучного інтелекту

Лабораторна робота №4

з дисципліни

«Дискретна математика»

Виконав:

студент групи КН-113

Сеньків Максим

Викладач:

Мельникова Наталя Іванівна

Львів – 2019 р.

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

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

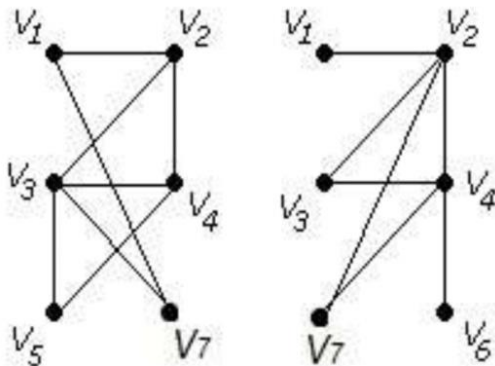
Варіант № 11

Завдання № 1.

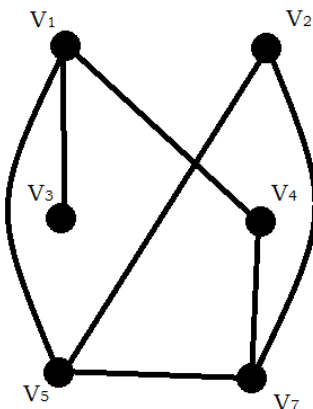
Розв'язати на графах наступні задачі:

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

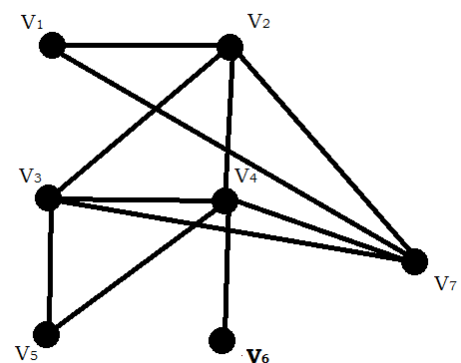
11



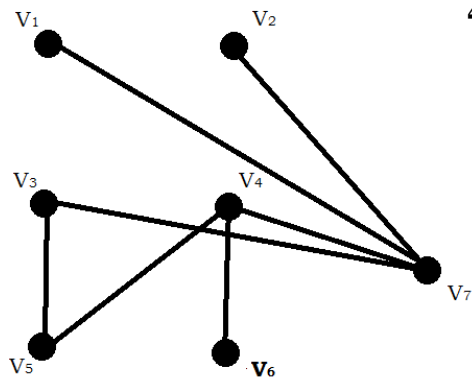
1)



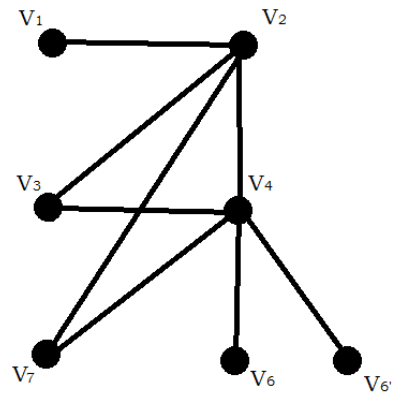
2)



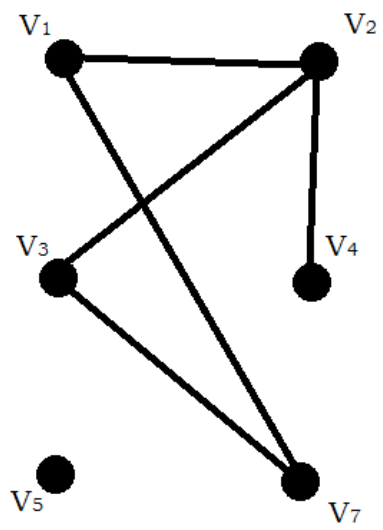
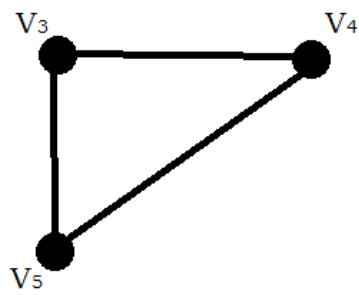
3)



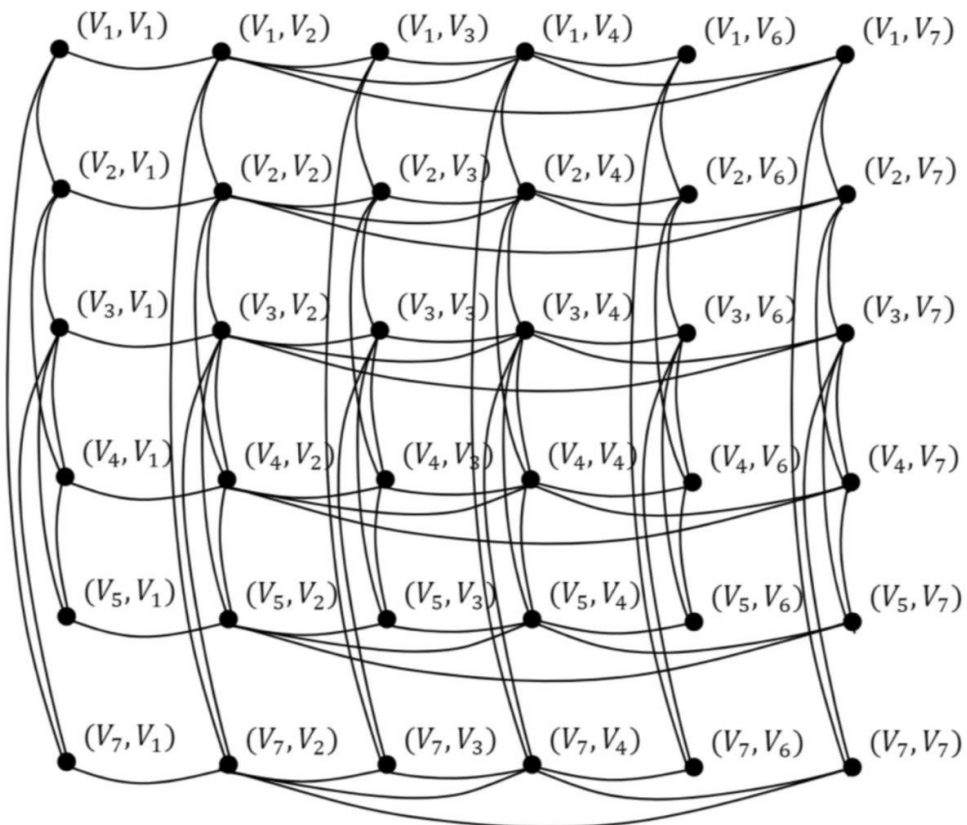
4)



5)

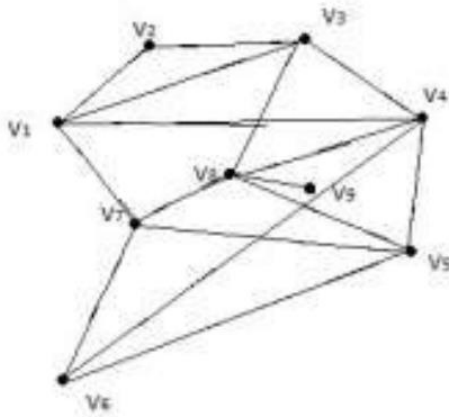


6)



Завдання №2

Знайти таблицю суміжності та діаметр графа.



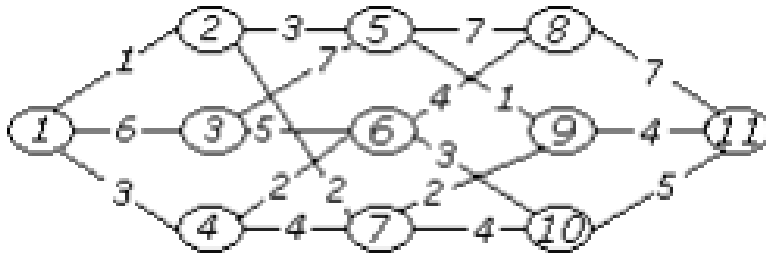
	V1	V2	V3	V4	V5	V6	V7	V8	V9
V1	0	1	1	1	0	0	1	0	0
V2	1	0	1	0	0	0	0	0	0
V3	1	1	0	1	0	0	0	1	0
V4	1	0	1	0	1	1	0	1	0
V5	0	0	0	1	0	1	1	1	0
V6	0	0	0	1	1	0	1	1	0
V7	1	0	0	0	1	1	0	1	0
V8	0	0	1	1	1	0	1	0	1
V9	0	0	0	0	0	0	0	1	0

Діаметр графа дорівнює 3, бо найдовша відстань між двома його вершинами не перевищує 3.

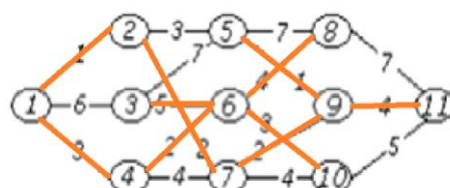
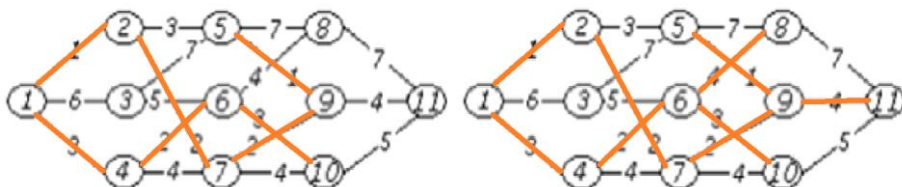
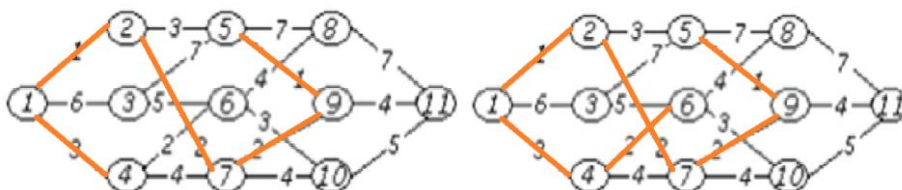
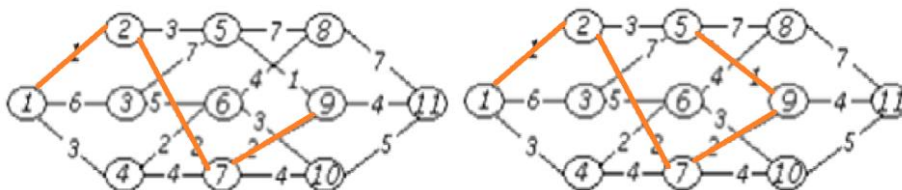
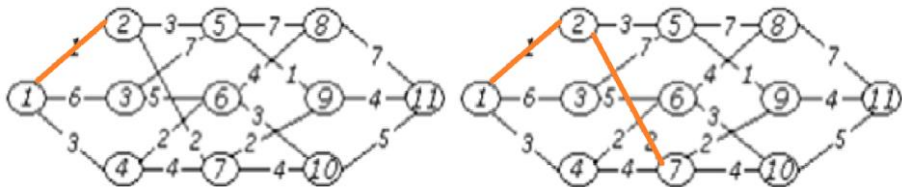
Завдання № 3

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

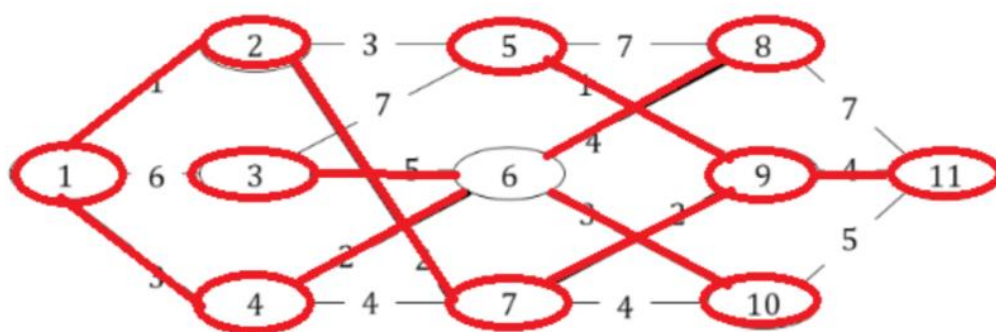
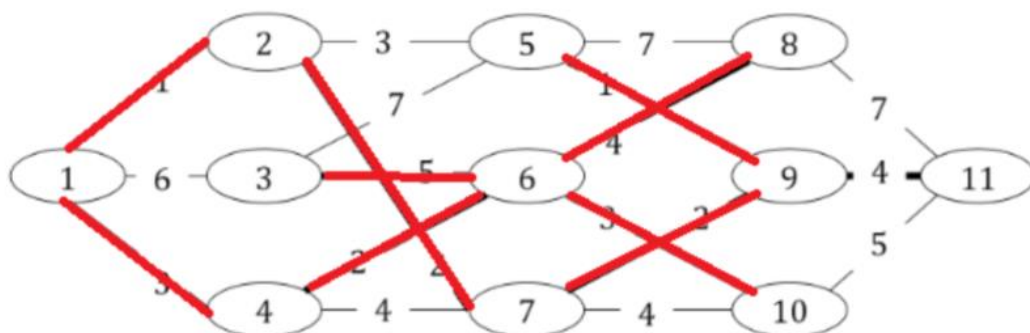
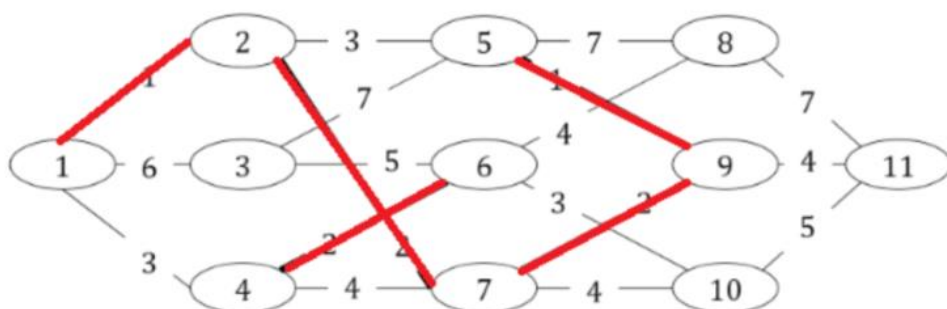
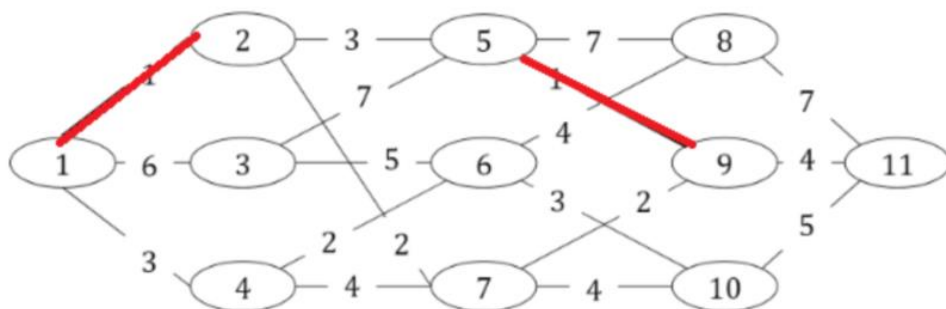
11



Алгоритм Прима:

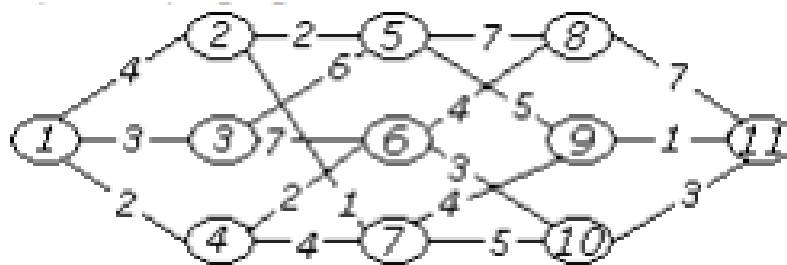


Алгоритм Краскала:



Частина №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 s=0;
5      int graf [11][11] {
6          {0,4,3,2,0,0,0,0,0,0,0},
7          {0,0,0,0,2,0,1,0,0,0,0},
8          {0,0,0,0,6,7,0,0,0,0,0},
9          {0,0,0,0,0,2,4,0,0,0,0},
10         {0,2,6,0,0,0,0,7,5,0,0},
11         {0,0,7,2,0,0,0,4,0,3,0},
12         {0,1,0,4,0,0,0,0,4,5,0},
13         {0,0,0,0,7,4,0,0,0,0,7},
14         {0,0,0,0,5,0,4,0,0,0,1},
15         {0,0,0,0,0,3,5,0,0,0,3},
16         {0,0,0,0,0,0,0,7,1,3,0},
17     };
18     int min(100), min1, min2, count(1);
19     int V[11];
20     V[0] = 0;
21     do {
22         for(int i = 0; i < count; i++) {
23             for (int x = 0; x < 11; x++) {
24                 if (graf[V[i]][x] != 0 && graf[V[i]][x] < min) {
25                     min = graf[V[i]][x];
26                     min1 = x;
27                     min2 = V[i];
28                 }
29             }
30         }
31         //знаходжу мінімальне ребро серед вершин, які відкриті
32         V[count] = min1;
```

```

33 // додаю до масиву наступну вершину
34 count++;
35 min = 10;
36 s+=graf[min2][min1];
37 for (int a = 0; a < 11; a++) {
38     graf[a][min1] = 0;
39 }
40 // зануляю стовбчик ребер з відкритою вершиною
41 cout << min2+1 << "-->" << min1+1 << "\t(" << min1 + 1 << ")" << endl;
42
43 }while (count < 11);
44
45 cout<<"Weight of graf is "<<s;
46 }

```

Виведення:

```

1-->4    (4)
4-->6    (6)
1-->3    (3)
6-->10   (10)
10-->11  (11)
11-->9   (9)
1-->2    (2)
2-->7    (7)
2-->5    (5)
6-->8    (8)
Weight of graf is 25

```

Висновки:

Я набув практичних навичок з використання алгоритмів Прима та Краскала.