**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**



**Лабораторна робота №5**

з дисципліни

«Дискретна математика»

**Виконав:**

студент групи КН-110

Король Орест

**Викладач:**

Мочурад Л.І.

Львів – 2017 рік

# *Лабораторна робота 5*

*****Тема:***** Знаходження найкоротошого маршруту за алгоритмом Дейкстри.

Плоскі планарні графи.

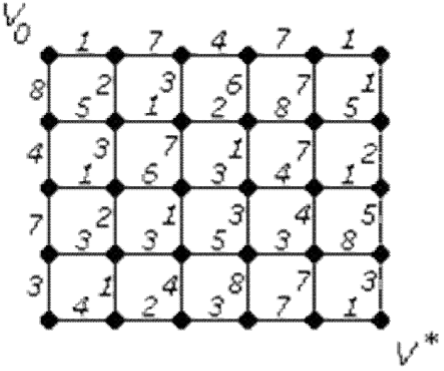
***Мета роботи:*** набуття практичних вмінь та навичок з використання

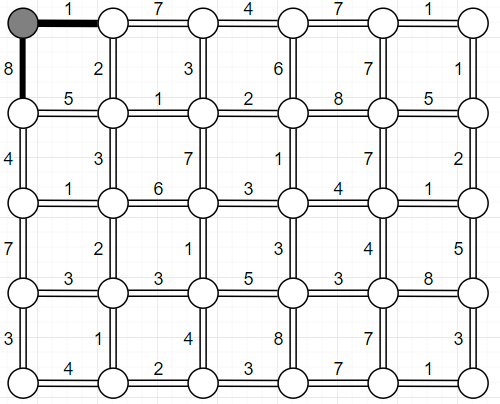
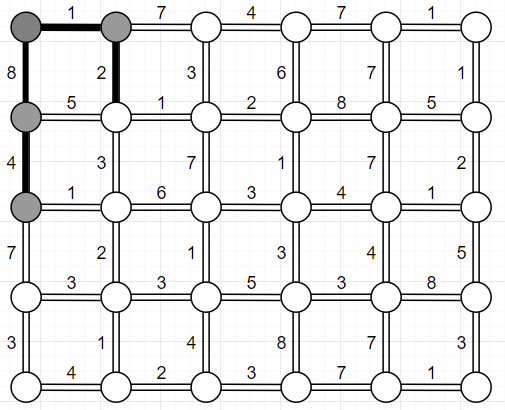
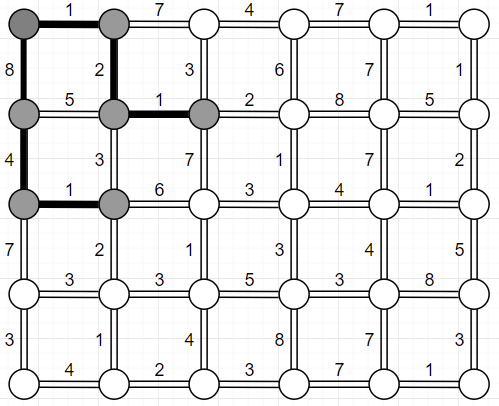
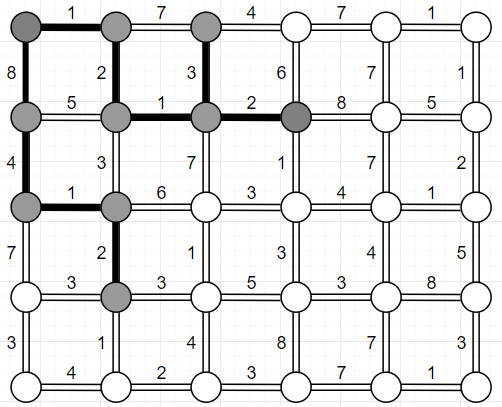
алгоритмів Дейкстри.

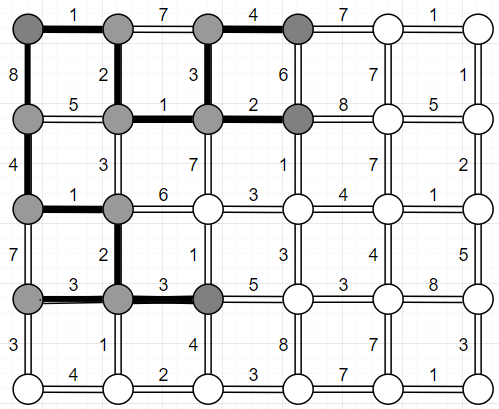
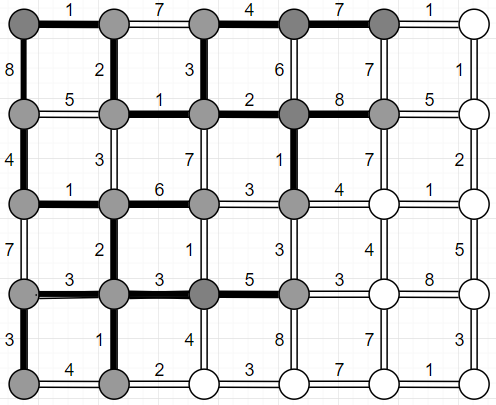
**Варіант №7**

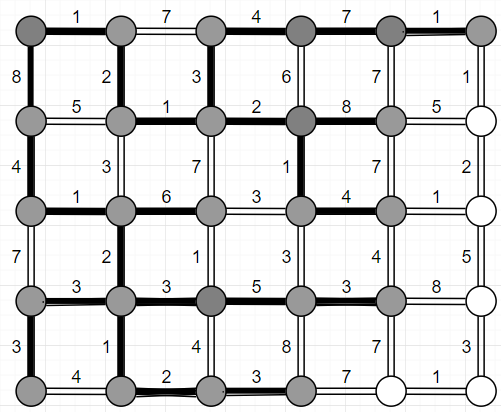
**Частина 1**

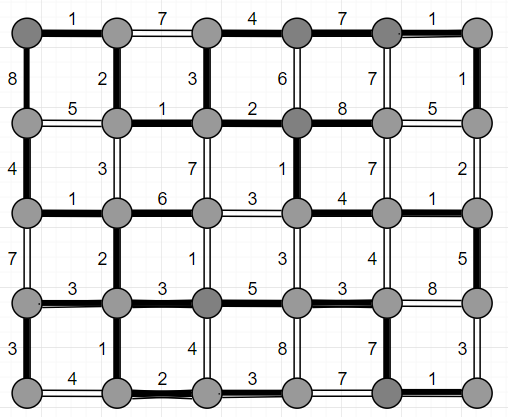
**Завдання № 1.** За допомогою алгоритму Дейкстра знайти найкоротший шлях у графі між нарою вершин V0 i V\*.

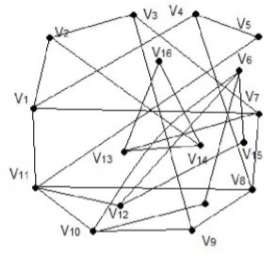




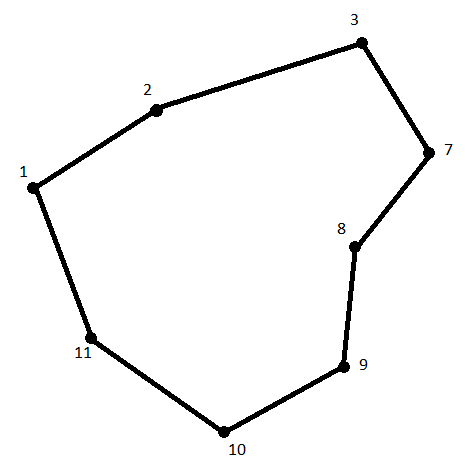


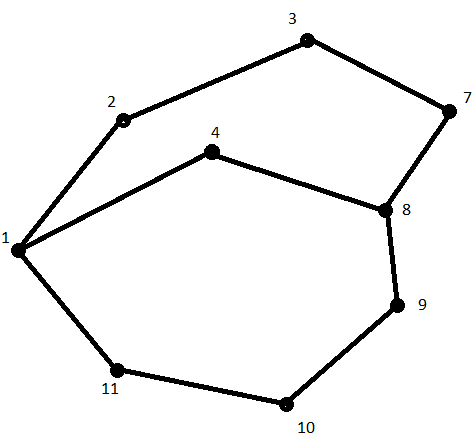
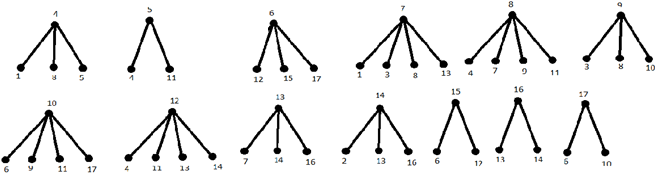


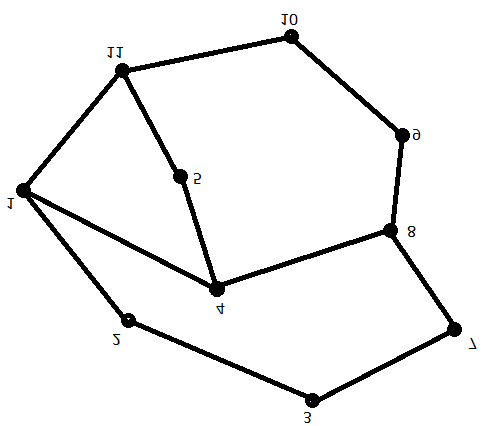
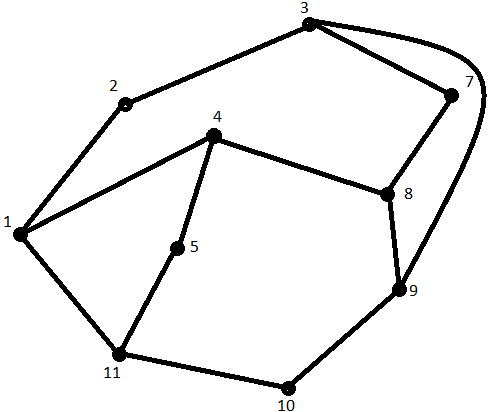
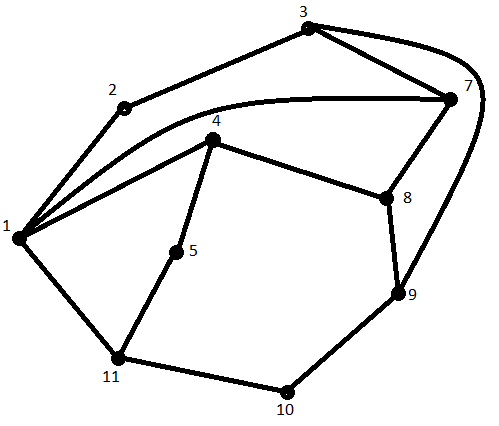


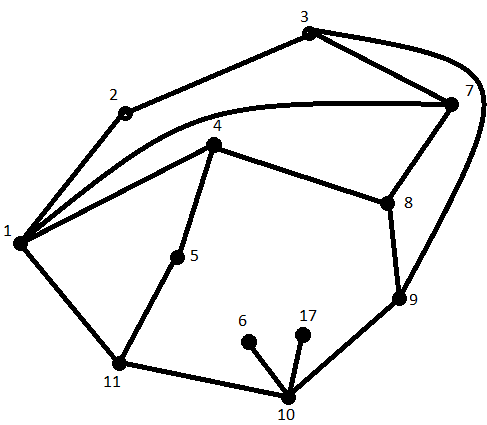
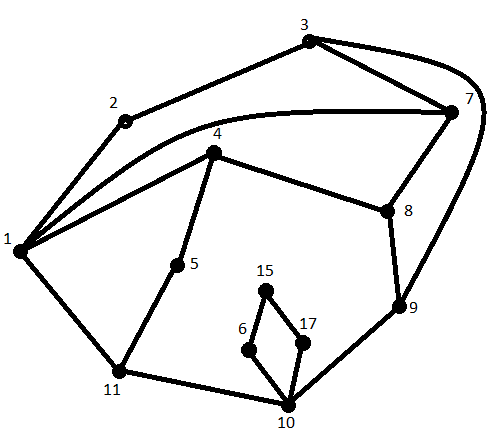
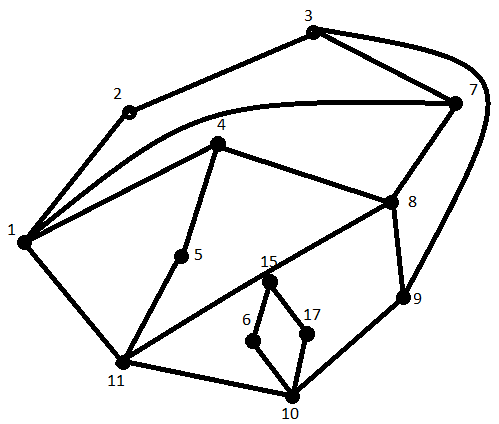
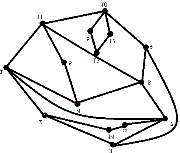
**Завдання №2:** За допомогою y-aлгоритма зробити укладку графа у площині.

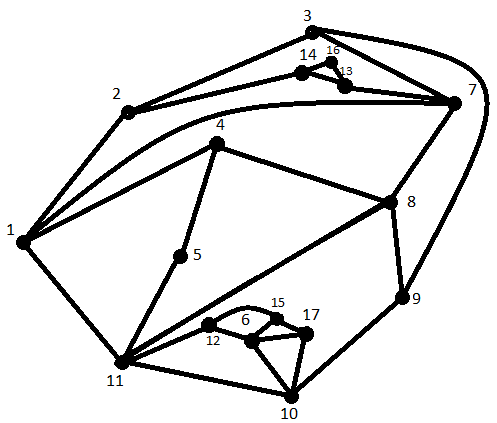
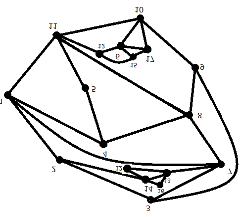
Виберемо цикл 1 – 2 – 3 – 7 – 8 – 9 – 10 – 11 - 1



****

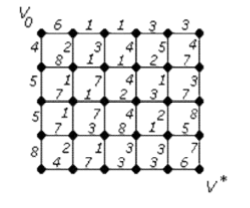






**Завдання №2**

Написати програму, яка реалізує алгоритм Дейкстри знаходження найкоротшого шляху між парою вершин у графі. Протестувати розроблену програму на наступному графі:



**Код програми:**

#include "stdafx.h"

#include<iomanip>

#include<iostream>

#include<string.h>

#include<stdio.h>

#include<stdlib.h>

#include<conio.h>

#pragma warning(disable : 4996)

using namespace std;

int i, j, n, p, xn, xk;

int flag[31];

int c[31][31], l[31];

char s[500], path[240][60];

int min(int n) {

int i, result;

for (i = 0; i<n; i++)

if (!(flag[i]))

result = i;

for (i = 0; i<n; i++)

if ((l[result]>l[i]) && (!flag[i]))

result = i;

return result;

}

int minim(int x,int y)

{

if (x<y)

return x;

return y;

}

void main()

{

setlocale(LC\_ALL, "ukr");

cout << "Введiть кiлькiсть точок: ";

cin >> n;

for (i = 0; i<n; i++)

for (j = 0; j<n; j++) c[i][j] = 0;

for (i = 0; i<n; i++)

for (j = i + 1; j<n; j++)

{

cout << "Введiть вiдстань вiд x" << i + 1 << " до x" << j + 1 << ": ";

cin >> c[i][j];

}

for (i = 0; i<n; i++)

{

for (j = 0; j<n; j++)

{

printf("%d ", c[i][j]);

c[j][i] = c[i][j];

}

printf("\n");

}

for (i = 0; i<n; i++)

for (j = 0; j<n; j++)

if (c[i][j] == 0)

c[i][j] = 65535;

cout << "Введiть початковy точкy: ";

cin >> xn;

cout << "Введiть кiнцевy точкy: ";

cin >> xk;

xk--;

xn--;

if (xn == xk)

{

cout << "Початкова i кiнцева точки збiгаються." << endl;

\_getch();

return;

}

for (i = 0; i<n; i++)

{

flag[i] = 0;

l[i] = 65535;

}

l[xn] = 0;

flag[xn] = 1;

p = xn;

\_itoa(xn + 1, s, 30);

for (i = 1; i <= n; i++)

{

strcpy(path[i], "X");

strcat(path[i], s);

}

do

{

for (i = 0; i<n; i++)

if ((c[p][i] != 65535) && (!flag[i]) && (i != p))

{

if (l[i] > l[p] + c[p][i])

{

if (i + 1 <= 9)

{

\_itoa(i + 1, s, 31);

strcpy(path[i + 1], path[p + 1]);

strcat(path[i + 1], "-X");

strcat(path[i + 1], s);

}

else if (i + 1 > 9 && i + 1 <= 19)

{

\_itoa(i - 9, s, 31);

strcpy(path[i + 1], path[p + 1]);

strcat(path[i + 1], "-X1");

strcat(path[i + 1], s);

}

else if (i + 1 > 19 && i + 1 <= 29)

{

\_itoa(i - 19, s, 31);

strcpy(path[i + 1], path[p + 1]);

strcat(path[i + 1], "-X2");

strcat(path[i + 1], s);

}

else if (i + 1 > 29)

{

\_itoa(i - 29, s, 31);

strcpy(path[i + 1], path[p + 1]);

strcat(path[i + 1], "-X3");

strcat(path[i + 1], s);

}

}

l[i] = minim(l[i], l[p] + c[p][i]);

}

p = min(n);

flag[p] = 1;

} while (p != xk);

if (l[p] != 65535)

{

cout << "Шлях: " << path[p + 1] << endl;

cout << "Довжина шляху: " << l[p] << endl;

}

else

cout << "Такого шляху не iснує" << endl;

\_getch();

}

**Bисновок:** я навчився виконувати алгоритм Дейкстри та укладати граф на площині.