Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №2**

По дисциплине «Математические основы интеллектуальных систем»

Тема: «Нахождение эйлеровых и гамильтоновых циклов в неориентированном графе»

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Карагодин Д.Л.

**Проверил:**

Козинский А.А.

Брест 2022

**Цель работы:** Изучение и решение заданий по построению таблицы смежности и инцидентности, нахождения компоненты связности неориентированного графа через поиск в ширину и глубину.

**Ход работы:**

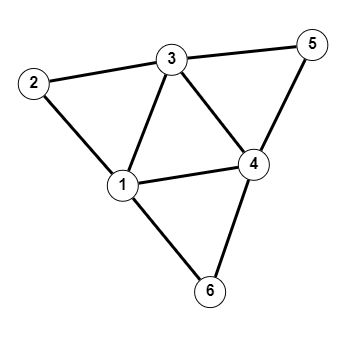
1. Написать программу нахождения эйлерова цикла в графе. Результатом программы является последовательность вершин в цикле. Предусмотреть проверку критерия на наличия эйлерова цикла в графе.
2. Написать программу нахождения гамильтонова цикла в графе. Результат программы – последовательность вершин цикла.
3. Изобразить графы.

**Вариант 4**

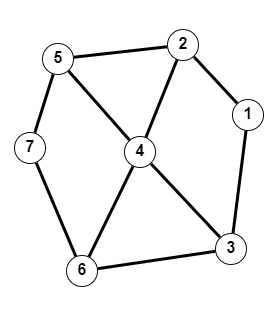
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Граф | *n* | Граф | *n* |
| 4. | (1,2),(1,3),(1,4),(1,6),(2,3),(3,4),  (3,5),(4,5),(4,6) | 6 | (1,3),(1,2),(2,4),(2,5),(3,4),(3,6),  (4,6),(4,5),(5,7),(6,7) | 7 |

**Изображение графов:**

**Граф **



**Граф** 



**Код программы:**

#include "1\_lab.hpp"

bool elier\_if(const Graphs& gh) {

vector<vector<bool>> adj = matrix\_adj(gh, false);

short count;

for (size\_t i = 0; i < gh.max; i++){

count = 0;

for (size\_t j = 0; j < gh.max; j++) if (adj[i][j]) count++;

if (!count % 2) { cout << "Incompatible elier graph" << endl; return false; }}

cout << "Compatible elier graph" << endl;

return true;}

void elier\_path(const Graphs& gh) {

vector<vector<bool>> adj = matrix\_adj(gh, false);

stack<short> st;

short i = gh.graph[0].first - 1;

st.push(i + 1);

bool flag;

while (true) {

flag = false;

for (size\_t j = 0; j < gh.max; j++){

if (adj[i][j]) {

adj[i][j] = 0; adj[j][i] = 0;

st.push(j + 1);

i = j;

flag = true;

break;}}

if (flag) continue;

cout << st.top();

st.pop();

if (!st.empty()) cout << "->";

else { cout << endl; break; }}}

void gamil\_cycle(const Graphs& gh) {

vector<int> path;

vector<vector<bool>> adj = matrix\_adj(gh, false);

short a=0,x;

bool no\_cont;

path.push\_back(a);

while (!path.empty()) {

x = path.back();

no\_cont = true;

for (size\_t i = 0; i < gh.max; i++){

if (adj[x][i]){bool in\_path = false;

for (size\_t j = 0; j < path.size(); j++) if (path[j] == i) { in\_path = true; break;}

if (!in\_path) {path.push\_back(i); no\_cont =false;adj[x][i]=0;break;}}}

if (no\_cont) {

for (size\_t m = 0; m < path.size(); m++)

if (path[m] == x) { path.erase(path.begin()+m);break;}

continue;}

if (path.size() == gh.max) {

if (adj[path.back()][a]) {

cout << "Gamil path: ";

for (auto const& m : path) cout << m + 1 << "->";

cout << a+1 << endl;

return;}}}}

int main() {

Graphs gh("graph.txt");

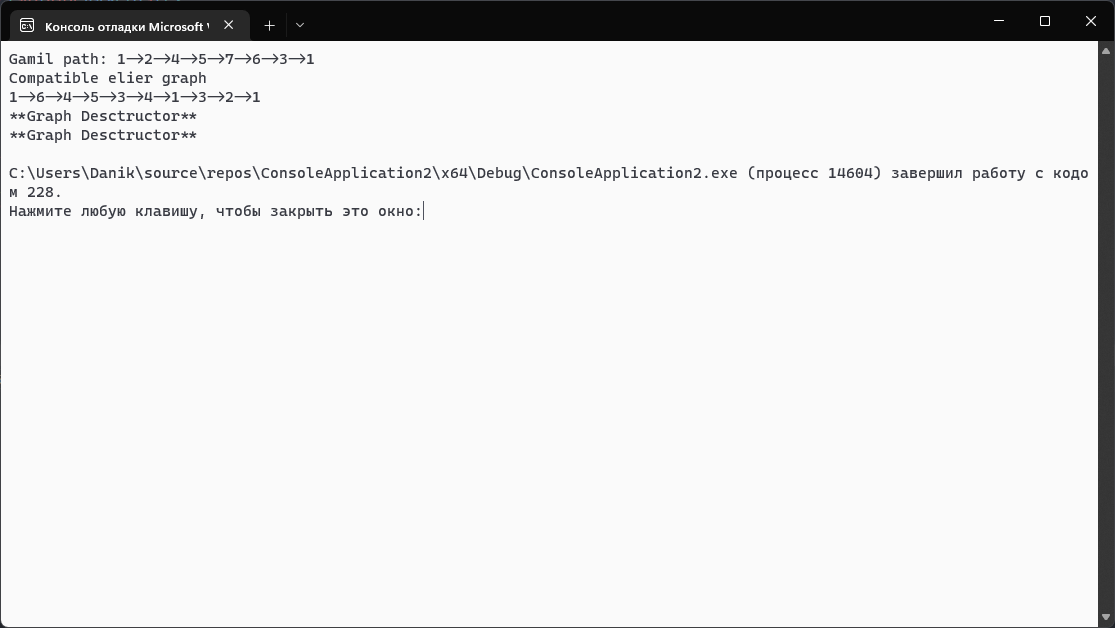
Graphs gh2("graph2.txt");

gamil\_cycle(gh2);

if (elier\_if(gh)) elier\_path(gh);

return 228;}

**Результат программы:**



**Вывод:** Изучил и решил задания нахождению эйлеровых и гамильтоновых циклов в неориентированном графе.