**МОЛДАВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет Математики и Информатики**

**Департамент Информатики**

Лабораторная работа № 3

(MDAG)

Проверил: профессор, др. N.Cuciuc

Выполнил: Иордатий Игнат

# Отчет по лабораторной работе

## Цель работы

Целью данной лабораторной работы является реализация алгоритма поиска максимального паросочетания в неориентированном графе с использованием алгоритма поиска в глубину (DFS).

## Теоретическая часть

Паросочетание в графе — это множество рёбер, не имеющих общих вершин. Максимальное паросочетание — паросочетание с максимальным возможным числом рёбер. В данной работе реализуется алгоритм поиска максимального паросочетания на основе жадного подхода и DFS.

## Исходный код программы

#include <iostream>  
#include <vector>  
  
class Graph {  
private:  
 int V;   
 std::vector<std::vector<int>> adj;   
  
public:  
 Graph(int vertices) {  
 V = vertices;  
 adj.resize(V);  
 }  
  
 void addEdge(int u, int v) {  
 adj[u].push\_back(v);  
 adj[v].push\_back(u);   
 }  
  
 bool dfs(int u, std::vector<bool>& visited, std::vector<int>& matching) {  
 for (int v : adj[u]) {  
 if (!visited[v]) {  
 visited[v] = true;  
 if (matching[v] == -1 || dfs(matching[v], visited, matching)) {  
 matching[v] = u;  
 return true;  
 }  
 }  
 }  
 return false;  
 }  
  
 int maxMatching() {  
 std::vector<int> matching(V, -1);  
 int result = 0;  
 for (int u = 0; u < V; u++) {  
 std::vector<bool> visited(V, false);  
 if (dfs(u, visited, matching)) {  
 result++;  
 }  
 }  
 return result / 2;  
 }  
};  
  
int main() {  
 int V, E;  
 std::cout << "Введите количество вершин: ";  
 std::cin >> V;  
 std::cout << "Введите количество рёбер: ";  
 std::cin >> E;  
  
 Graph g(V);  
  
 std::cout << "Введите " << E << " пар вершин (нумерация с 0): " << std::endl;  
 for (int i = 0; i < E; i++) {  
 int u, v;  
 std::cin >> u >> v;  
 g.addEdge(u, v);  
 }  
  
 std::cout << "Максимальное количество паросочетаний: " << g.maxMatching() << std::endl;  
  
 return 0;  
}

## Пример работы программы

Введите количество вершин: 4  
Введите количество рёбер: 4  
Введите 4 пар вершин (нумерация с 0):  
0 1  
1 2  
2 3  
3 0  
Максимальное количество паросочетаний: 2

## Вывод

В ходе лабораторной работы была реализована структура данных "граф", реализована функция поиска максимального паросочетания с помощью DFS. Полученные результаты подтверждают корректность реализации алгоритма.