Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №4**

По дисциплине «Математические основы интеллектуальных систем»

Тема: «Группы»

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Литвинюк Т. В.

**Проверил:**

Козинский А. А.

Брест 2022

**Цель:** научиться основным понятиям комбинаторики.

**Ход работы:**

**Вариант 7**

****

#include <iostream>

#include <fstream>

#include <windows.h>

using namespace std;

int \*get\_substitution(string path){

ifstream file(path);

char numbers[100];

file >> numbers;

int \*substitution, i = 1, t = 1, n = 0, m = 0;

char temp[] = "00";

substitution = new int[6];

while (i < 50){

while (numbers[i] != ',' && numbers[i] != ']')

temp[t--] = numbers[i++];

substitution[n++] = (temp[0] - 48) \* 10 + temp[1] - 48;

temp[0] = '0';

t = 1;

if (numbers[i] == ']')

break;

i++;

}

return substitution;

}

// 11111111111111111111111111111111111111111111111111

void independentСycles(int \*substitution, int \*part1, int \*part2){

//независимые циклы

int flag;

int p1 = 1, p2 = 1;

part1[0] = 1;

while (p1 < 7 && part1[0] != substitution[part1[p1 - 1] - 1]){

part1[p1++] = substitution[part1[p1 - 1] - 1];

}

for (int i = 1; i < 7; i++){

flag = 0;

for (int j = 0; j < p1; j++)

if (part1[j] == i)

flag = 1;

if (!flag){

part2[0] = i;

break;

}

}

while (p2 < 7 && part2[0] != substitution[part2[p2 - 1] - 1]){

part2[p2++] = substitution[part2[p2 - 1] - 1];

}

cout << "(";

for (int i = 0; i < p1; i++)

cout << part1[i] << ",";

cout << "\b)(";

for (int i = 0; i < p2; i++)

cout << part2[i] << ",";

cout << "\b)\n";

}

// 22222222222222222222222222222222222222222222222222

int numberOfTranspositions(int \*substitution){

//количество перестановок

int count = 0;

for (int i = 0; i < 6; i++)

if (substitution[i] != i + 1)

count++;

return count;

}

// 33333333333333333333333333333333333333333333333333

int numberOfInversions(int \*substitution){

//количество инверсий

int inversions = 0;

for (int i = 0; i < 6; i++)

for (int j = i + 1; j < 6; j++)

if (substitution[i] > substitution[j])

inversions++;

return inversions;

}

int main(){

setlocale(LC\_ALL, "Russian");

int \*a = get\_substitution("for\_task1.txt");

int part1[6], part2[6];

independentСycles(a, part1, part2);

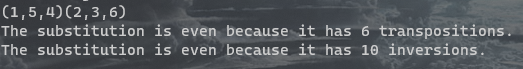
int transpositions = numberOfTranspositions(a);

cout << "The substitution is " << (transpositions % 2 == 0 ? "even" : "odd") << " because it has " << transpositions << " transpositions." << endl;

int inversions = numberOfInversions(a);

cout << "The substitution is " << (transpositions % 2 == 0 ? "even" : "odd") << " because it has " << inversions << " inversions." << endl;

}

****

**Вывод:** в ходе лабораторной работы я научился находить кратчайшие пути в графе.