Министерство образования и науки РФ

ФГАОУ ВПО «Уральский федеральный университет имени   
первого президента России Б.Н.Ельцина»  
Институт радиоэлектроники и информационных технологий – РТФ  
Департамент информационных технологий и автоматики

Оценка работы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ЗАДАЧА О РАССТАНОВКЕ ФЕРЗЕЙ**

Отчет по лабораторной работе №4  
по дисциплине «Методы исследования и моделирования  
информационных процессов и технологий»

Подпись Дата Ф.И.О.

Преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Добряк П.В.  
Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тимошенкова Ю.С.

Группа РИМ-161203

Екатеринбург

2017

**Задача:** вводится размер квадратного поля - количество клеток по стороне *n*. Нужно расставить *n* ферзей так, чтобы они друг друга не порубили (ферзь рубит всех по своей строке, столбцу и диагоналям). Итог работы программы – количество возможных расстановок и пример одной расстановки.

**Описание принципа решения задачи:**

Заполнить матрицу размером 8×8 нулями и единицами таким образом, чтобы сумма всех элементов матрицы была равна 8, при этом сумма элементов ни в одном столбце, строке или диагональном ряде матрицы не превышала единицы

Таблица, зависимость количества решений от размера доски:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| n | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Решений | 0 | 0 | 2 | 10 | 4 | 40 | 92 | 352 | 724 |

**Алгоритм поиска с возвратом:**

1. Начинаем с пустой доски.
2. Ставим фигурку в первую свободную клетку:
   1. Если фигурка атакует другую, тогда переставляем ее на следующую свободную клетку:
      1. Если больше нет не попробованных клеток, тогда возвращаемся к предыдущей установленной фигурке и переставляем ее:
         1. Но если текущая фигурка является начальной, тогда предыдущей быть не может, следовательно, задача не имеет решения.
   2. Если количество фигурок равно размерности доски тогда данная доска является решением.
   3. Если количество фигурок меньше размерности доски тогда берем еще одну фигурку и возвращаемся к пункту 2.

**Код реализации на языке C++**

#include "stdafx.h"

#include<iostream>

#include <stdio.h>

#include <tchar.h>

#include <math.h>

using namespace std;

const int N = 9; // Размер доски и количество ферзей

int X[N]; // Шахматная доска

int Count; // Количество расстановок

// Функция поиска свободных клеток на доске

bool Search\_Position(int X[N], int v, int g)

{

/\*

v - вертикаль

g - горизонталь

Описание условия while ((i<k) && (y != X[i]) && (abs(k - i) != abs(y - X[i]))) { i++; }

(i<k) - проверка предыдущих вертикалей

(y != X[i]) - проверка горизонтали

(abs(k - i) != abs(y - X[i])) - проверка диагонали

\*/

int i = 0;

while ((i<v) && (g != X[i]) && (abs(v - i) != abs(g - X[i]))) { i++; }

if (i == v)

return true;

else if (i != v)

return false;

else

return !true && !false;

}

// Функция поиска с возвратом

void Search\_Backtracking(int k)

{

int i, y;

for (y = 0; y<N; y++)

if (Search\_Position(X, k, y))

{

X[k] = y;

if (k == N - 1)

{

Count++;

}

Search\_Backtracking(k + 1);

}

}

// Функция вывода всех возможных позиций ферзей

void Vivod\_Position(int k)

{

int i, y;

for (y = 0; y<N; y++)

if (Search\_Position(X, k, y))

{

X[k] = y;

if (k == N - 1) {

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

{

cout << char('A' + i) << X[i] + 1 << " ";

}

cout << endl;

}

Vivod\_Position(k + 1);

}

}

int main()

{

for (int i = 0; i<N; i++)X[i] = 0;

Count = 0;

cout << "Zadacha rasstavit " << N << " ferzey:" << endl;

Search\_Backtracking(0);

cout << "Na doske razmerom " << N << " na " << N << " mojno rasstavit " << Count << " sposobami " << endl;

cout << "Na doske razmerom " << N << " na " << N << " primer rasstanovki: " << endl;

Vivod\_Position(0);

cout << endl;

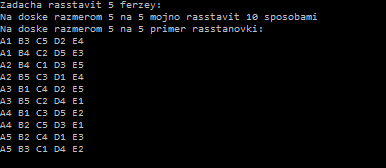
system("pause");

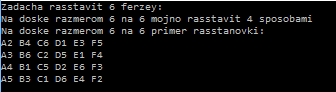
}

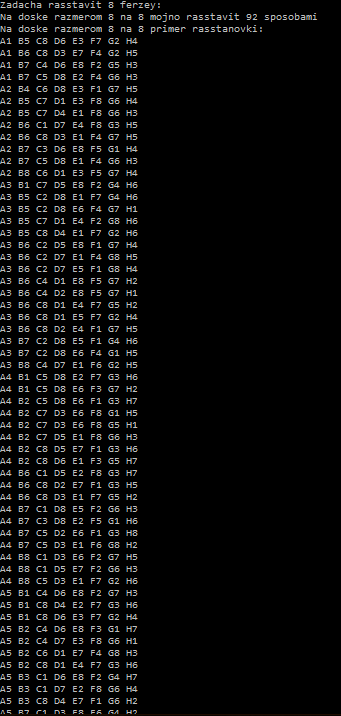
Результат работы алгоритма

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| n | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Решений ожидаемо | 0 | 0 | 2 | 10 | 4 | 40 | 92 | 352 | 724 |
| Решений получено | 0 | 0 | 2 | 10 | 4 | 40 | 92 | 352 | 724 |









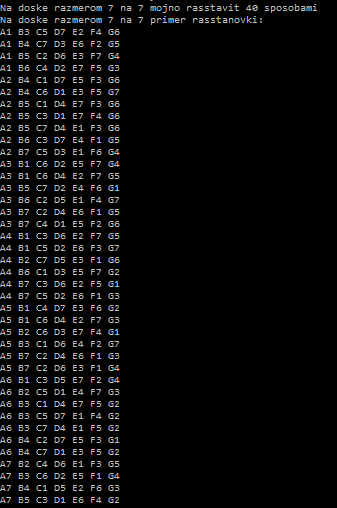


Рисунок 1 – Результат работы программы

# ВЫВОДЫ

В ходе выполнения лабораторной работы был реализован алгоритм расстановки n ферзей на шахматной доске размером n на n таким образом, чтобы ни один из них не бил другого. По результатам выполнения работы представлена таблица сравнения ожидаемого результата и полученного результата, полученные результаты совпадают, что говорит о правильности реализации алгоритма.