Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

**по лабораторной работе №13**

Дисциплина: основы алгоритмизации и программирования

Тема: Рекурсивные функции. Задача о расстановке 8восьми ферзей

Вариант: NULL

Выполнил работу

студент группы РИС-20-1б

Рябов Никита Андреевич

Проверила

Доцент кафедры ИТАС Полякова О.А.

Пермь, 2021

**Постановка задачи**

Решить задачу о 8 ферзях методом поиска с возвратом

**Анализ задачи**

*Какие предстоит выполнить действия.*

Заданный как глобальная переменная массив заполняется нулями

for (int i = 0; i < 8; i++) for (int j = 0; j < 8; j++) board[i][j] = 0;

Вызывается рекурсивная функция расстановки ферзей

try\_queen(0);

Обработанный массив выводится на печать

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 8; j++)

{

if (board[i][j] == -1) cout << "Y";

else cout <<".";

}

cout << endl;

}

Функция try\_queen ставит ферзя на первую свободную клетку заданной строки. Вызывает саму себя для следующей строки. Если вызванная функция возвращает false, то она не смогла поставить ферзь на следующую строку. В этом случае функция верхнего уровня убирает ферзь с доски и ставит ее на следующую свободную от боя клетку.

bool try\_queen(int i)

{

bool result = false;

for (int j = 0; j < 8; j++)

{

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

{

SR\_queen(i, j, 1, -1);

if (i == 7)

result = true;

else if (!(result = try\_queen(i + 1))) SR\_queen(i, j, -1, 0);

}

if (result) break;

}

return result;

}

Данная функция отвечает за постановку/убирание ферзей. Она проходит по диагоналям, вертикалям и горизонталям. Если ферзь ставится, прибавляет 1, иначе вычитает. В конце ставится сам ферзь, то есть -1.

void SR\_queen(int i, int j, int add, int queen)

{

int foo;

for (int x = 0; x < 8; x++)

{

board[x][j]+=add;

board[i][x]+=add;

foo = j - i + x;

if (foo >= 0 && foo < 8)

board[x][foo]+=add;

foo = j + i - x;

if (foo >= 0 && foo < 8)

board[x][foo]+=add;

}

board[i][j] = queen;

}

*С каким типом данных действуем*

Целые числа

*В каком виде представлены данные*

Двумерный массив целых чисел

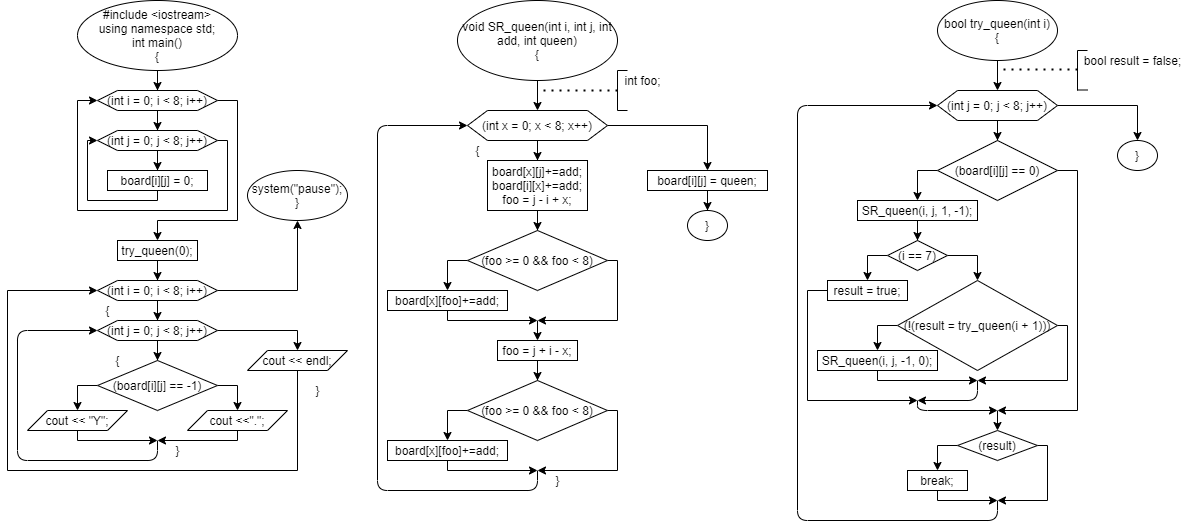
*Какие поля*

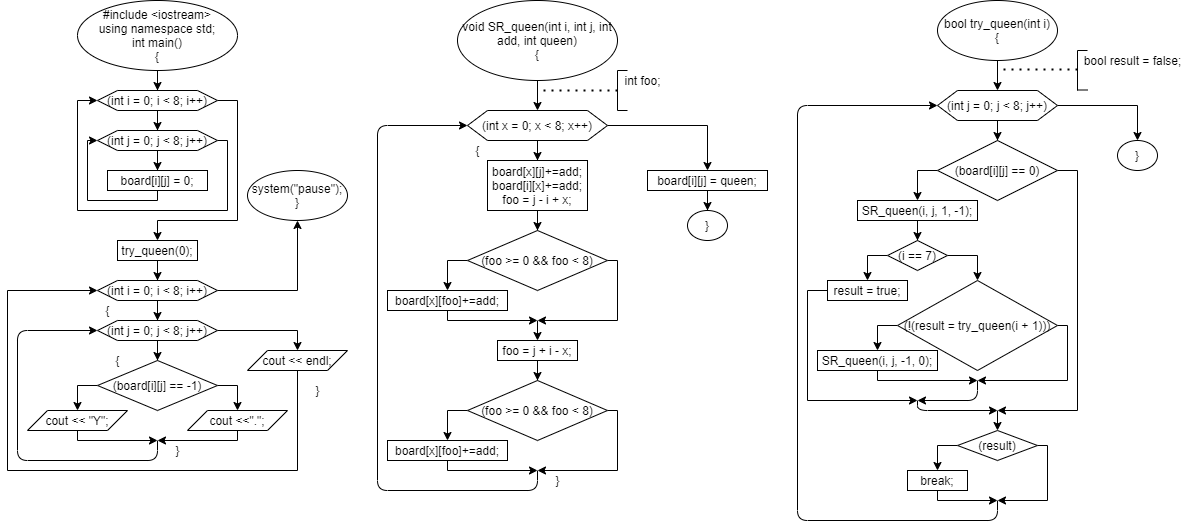
Структуры в данной программе отсутствуют

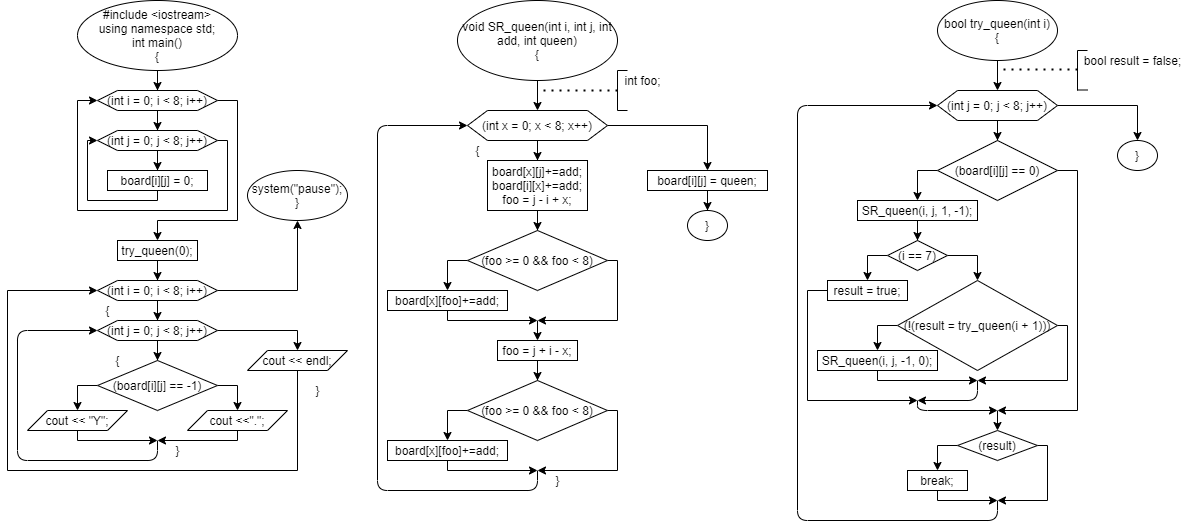
*Операторы ввода/вывода*

Для вывода данных используется команда cout. Ввод программе не требуется

**Блок-схема**

****

****

****

**Код**

#include <iostream>

using namespace std;

int board[8][8];

void SR\_queen(int i, int j, int add, int queen)

{

int foo;

for (int x = 0; x < 8; x++)

{

board[x][j]+=add;

board[i][x]+=add;

foo = j - i + x;

if (foo >= 0 && foo < 8)

board[x][foo]+=add;

foo = j + i - x;

if (foo >= 0 && foo < 8)

board[x][foo]+=add;

}

board[i][j] = queen;

}

bool try\_queen(int i)

{

bool result = false;

for (int j = 0; j < 8; j++)

{

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

{

SR\_queen(i, j, 1, -1);

if (i == 7)

result = true;

else if (!(result = try\_queen(i + 1))) SR\_queen(i, j, -1, 0);

}

if (result) break;

}

return result;

}

int main()

{

for (int i = 0; i < 8; i++) for (int j = 0; j < 8; j++) board[i][j] = 0;

try\_queen(0);

for (int i = 0; i < 8; i++)

{

for (int j = 0; j < 8; j++)

{

if (board[i][j] == -1) cout << "Y";

else cout <<".";

}

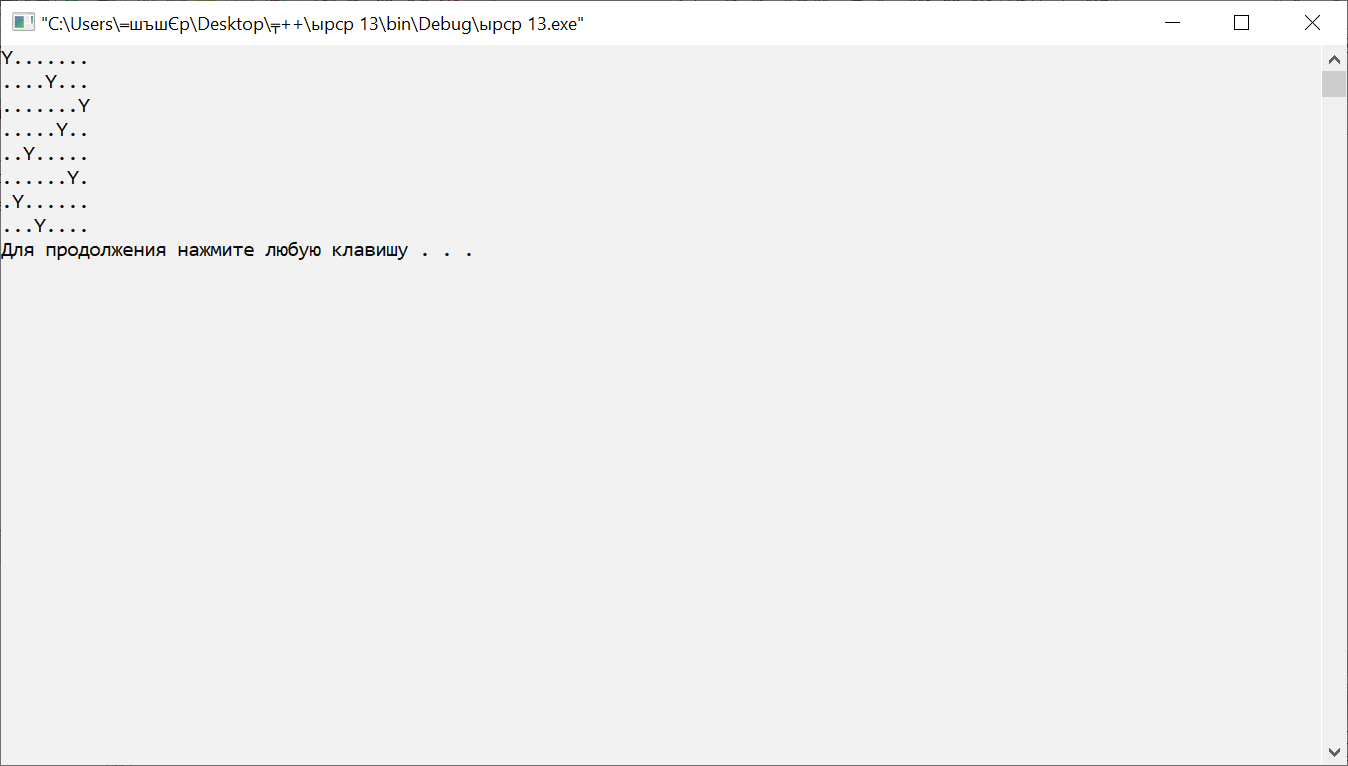
cout << endl;

}

system("pause");

}

**Скриншоты**

****