Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное учреждение высшего образования «Пермский национальный исследовательский политехнический университет»

ПНИПУ

Лабораторная работа №6  
«8 ферзей»

Выполнил:   
студент группы РИС-23-1б   
Мокрушин Никита Дмитриевич

Проверила:   
доцент кафедры ИТАС   
О.А. Полякова

Пермь 2024 г.

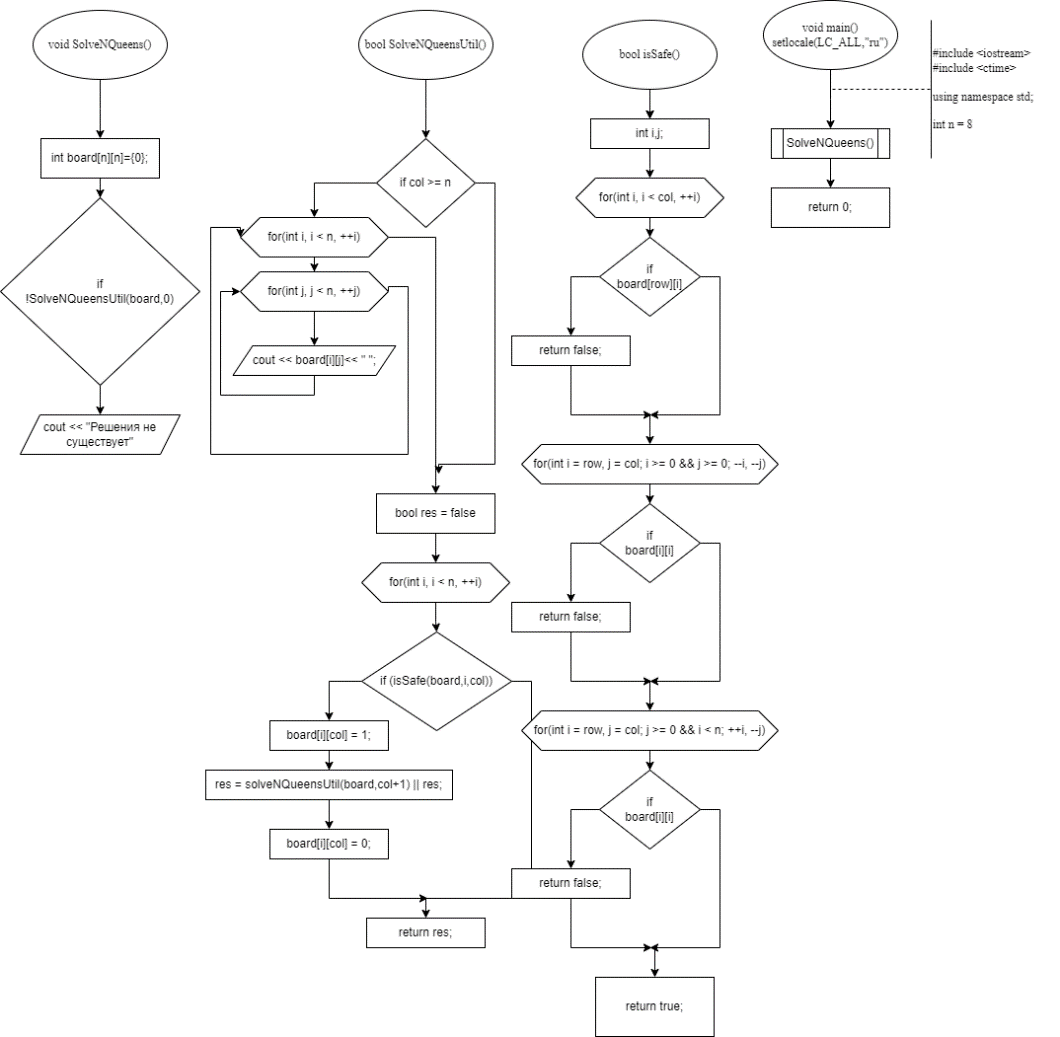
**Постановка задачи:**

Реализовать с помощью рекурсии задачу о восьми ферзях

**Анализ задачи «8 ферзей»:**

* Создаем функцию для проверки можно ли разместить ферзя на данной позиции
* Смотрим доступность по вертикали, горизонтали и диагоналям
* Создаем саму рекурсивную функцию для решения задачи
* В начале выводим текущее состояние доски
* Размещаем ферзя в текущем столбце для каждой строки, проверяем на размещение ферзя
* Если размещение невозможно возвращаем доску к исходому состоянию
* В функции void main() вызываем рекурсивную функцию

**Блок-схема:**

****

**Код на языке с++:**

#include <iostream>

using namespace std;

const int N = 8;

bool isSafe(int board[N][N], int row, int col)

{

int i, j;

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

if (board[row][i])

return false;

for (i = row, j = col; i >= 0 && j >= 0; --i, --j)

if (board[i][j])

return false;

for (i = row, j = col; j >= 0 && i < N; ++i, --j)

if (board[i][j])

return false;

return true;

}

bool solveNQueensUtil(int board[N][N], int col)

{

if (col >= N) {

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

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

cout << board[i][j] << " ";

cout << endl;

}

cout << endl;

return true;

}

bool res = false;

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

{

if (isSafe(board, i, col))

{

board[i][col] = 1;

res = solveNQueensUtil(board, col + 1) || res;

board[i][col] = 0;

}

}

return res;

}

void solveNQueens() {

int board[N][N] = { 0 };

if (!solveNQueensUtil(board, 0))

cout << "Решение не существует";

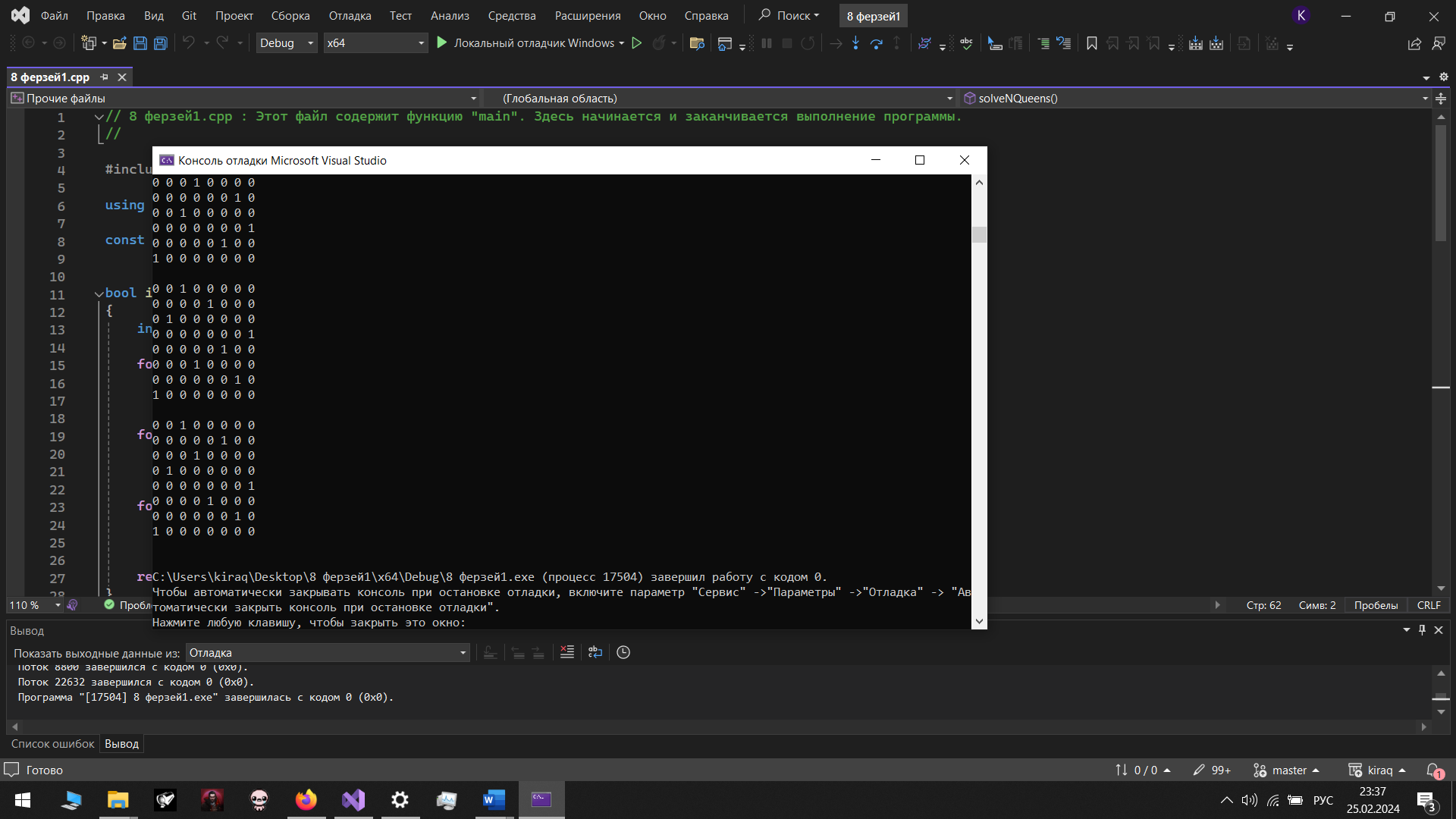
}

int main() {

solveNQueens();

return 0;

**Пример работы программы:**



**Вывод:** Программа работает как надо