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

ПНИПУ

**Лабораторная работа  
“Задача**

**о восьми Ферзях ”**

Выполнил:   
студент группы РИС-23-1б   
Никонов Антон Сергеевич

Проверила:   
доцент кафедры ИТАС   
О.А. Полякова

Пермь, 2023 г.

**Цель:**

Разработать программу, позволяющая решить задачу о восьми ферзях – расставить их на шахматной доске 8 на 8 таким образом, чтобы ни один ферзь не атаковал другого

**Анализ задачи:**

Ферзь может двигаться в любом направлении по горизонтали, вертикали, диагонали и на любое количество клеток, он также может "рубить" фигуры, как и ходить. Чтобы ферзи не "били" друг друга, на каждой строке, диагонали и каждом столбце должен находиться только один ферзь.

Для расстановки ферзей требуется:

1. Поставить первого ферзя на позицию а1 (первая клетка первой строки).
2. Перейти на следующую строку и поставить ферзя так, чтобы первый ферзь его не бил.
3. Если на какой-либо строке поставить ферзя невозможно(так, чтобы они не «били» друг друга), то возвращаемся на предыдущую строку и ставим ферзя на следующую клетку строки.
4. Повторяем пункты 2 и 3, пока не расставим всех ферзей.

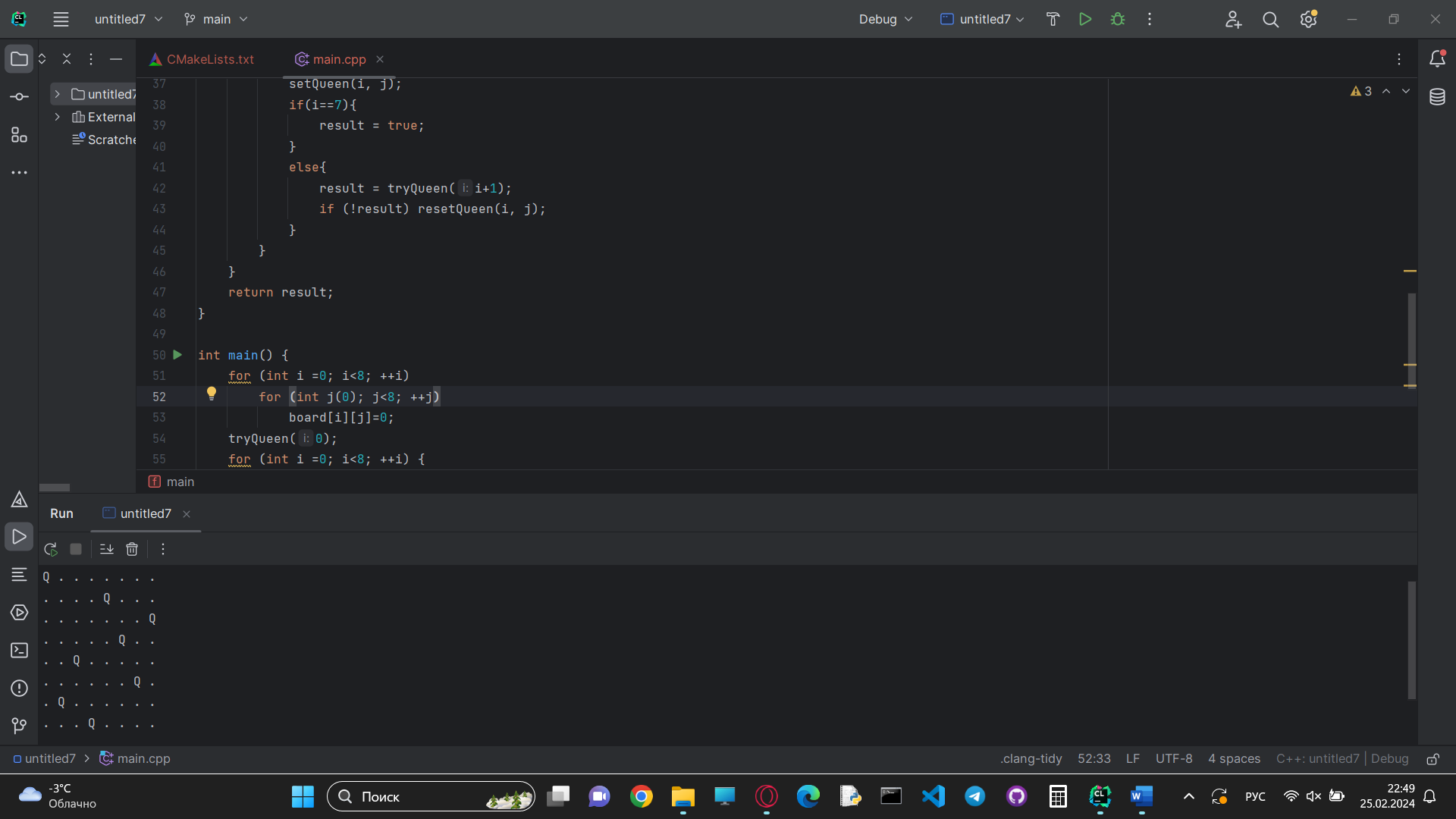
Алгоритм решения:

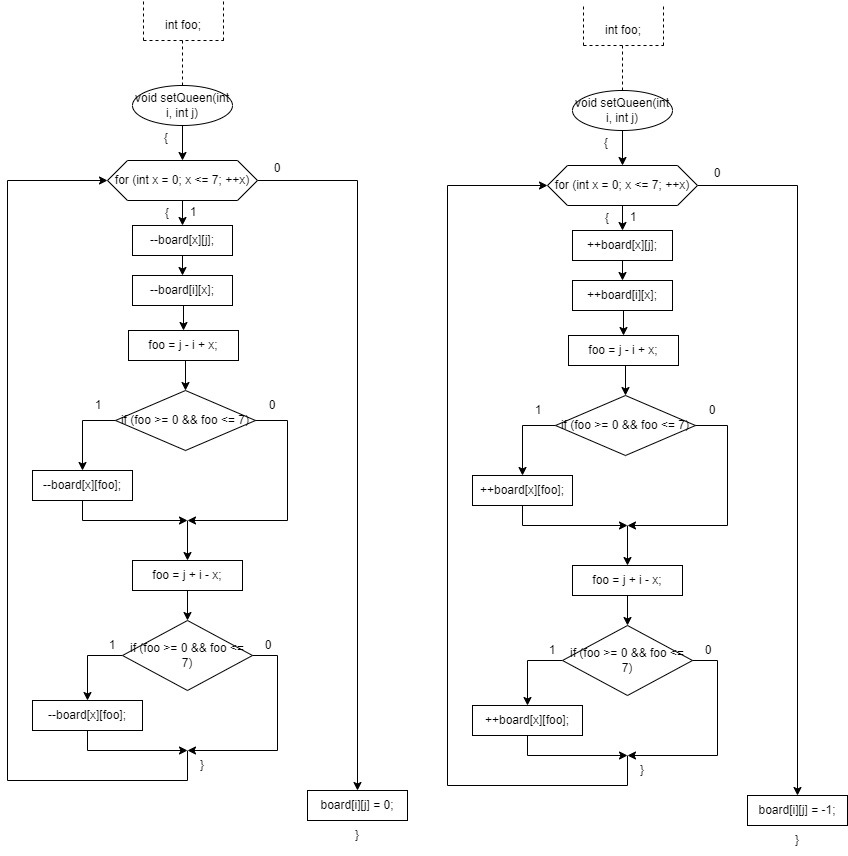
1. Необходимо реализовать две функции: поставить ферзя и убрать ферзя. Первая будет ставить в заданную клетку ферзя, а во все остальные отмечать, что они находятся под боем. Вторая аналогично будет убирать ферзя и клетки шахматной доски, находящиеся под боем.
2. Помимо этого должна быть функция, которая выбирает, куда поставить ферзей, она поочередно проходит строки и ставит ферзей так, чтобы они находились не под боем(ферзи не должны находиться в одной строке, в одном столбце и на одной диагонали). Но если на какой-то строке функция уже не может поставить ферзя, то она возвращается на шаг назад, убирает предыдущего ферзя и пытается поставить его на другое место. И так до тех пор, пока все ферзи не займут свои места.

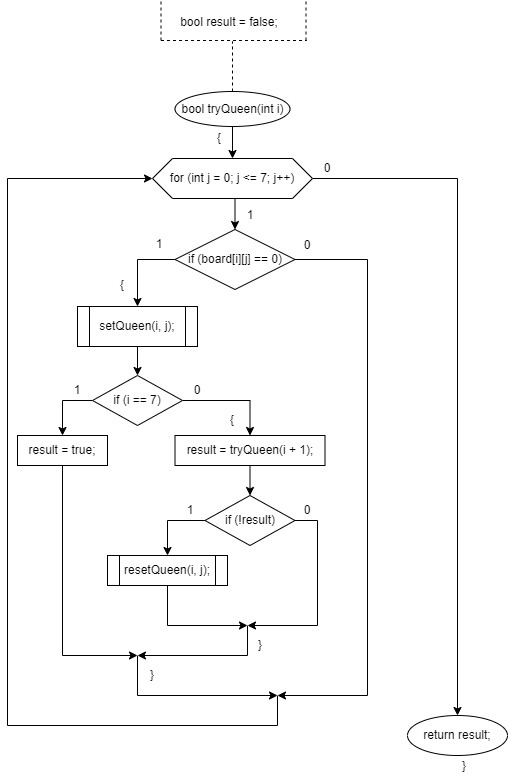
**Код программы:**

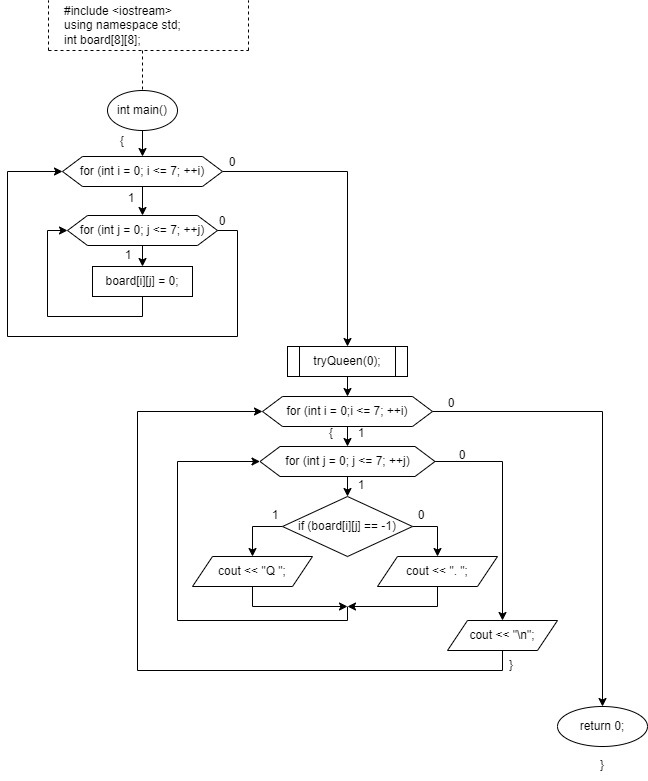
#include <iostream>  
using namespace std;  
  
int board[8][8];  
  
void setQueen(int i, int j)  
{  
 for (int x(0); x < 8; x++){  
 ++board[x][j];  
 ++board[i][x];  
 int foo;  
 foo = j - i + x;  
 if (foo >= 0 && foo < 8) ++board[x][foo];  
 foo = j+i-x;  
 if (foo >= 0 && foo < 8) ++board[x][foo];  
 }  
 board[i][j] = -1;  
}  
void resetQueen(int i, int j)  
{  
 for (int x(0); x < 8; x++){  
 --board[x][j];  
 --board[i][x];  
 int foo;  
 foo = j - i + x;  
 if (foo >= 0 && foo < 8) --board[x][foo];  
 foo = j+i-x;  
 if (foo >= 0 && foo < 8) --board[x][foo];  
 }  
 board[i][j] = 0;  
}  
  
bool tryQueen(int i){  
 bool result = false;  
 for (int j(0); j < 8; j++){  
 if (board[i][j]==0){  
 setQueen(i, j);  
 if(i==7){  
 result = true;  
 }  
 else{  
 result = tryQueen(i+1);  
 if (!result) resetQueen(i, j);  
 }  
 }  
 }  
 return result;  
}  
  
int main() {  
 for (int i =0; i<8; ++i)  
 for (int j(0); j<8; ++j)  
 board[i][j]=0;  
 tryQueen(0);  
 for (int i =0; i<8; ++i) {  
  
 for (int j(0); j < 8; ++j) {  
 if (board[i][j] == -1) cout << "Q ";  
 else cout << ". ";  
 }  
 cout << endl;  
 }  
 return 0;  
}

**Вывод:**

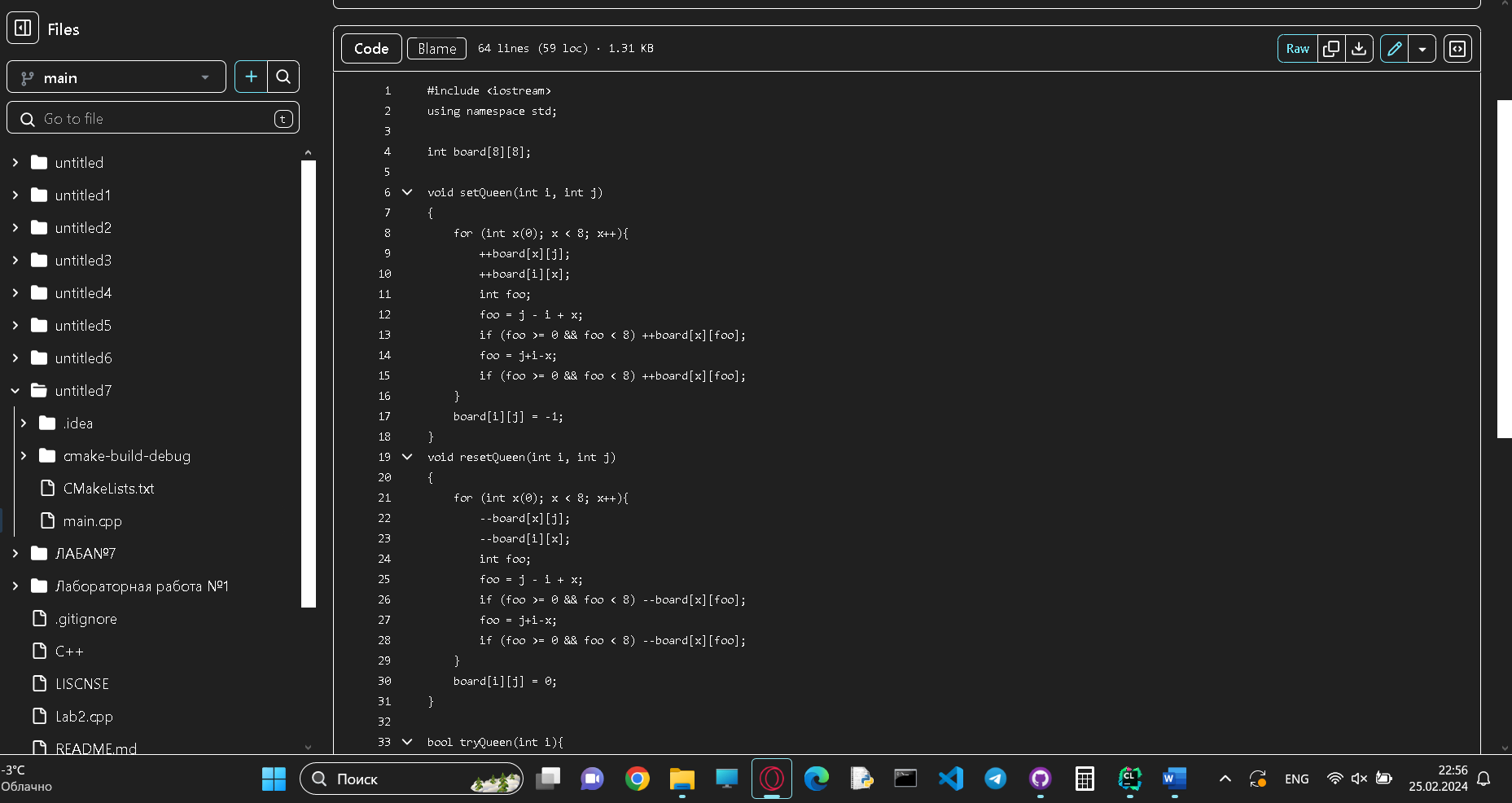
****

**Блок-схем****а:**





**Скриншоты с гита:**

****

**Вывод:**

Программа выполняет свою задачу.