Министерство общего и профессионального

образования Российской Федерации

Пермский государственный технический университет

Лабораторная работа

"Задача о восьми ферзях"

Выполнила студентка группы РИС 23-3б:

Федорова О.И.

Пермь 2024

1. Постановка задачи. Дано: шахматная доска – 64 клетки. Нужно расставить 8 ферзей: по одному в каждую строку так, чтобы никакой из ферзей не бил другого. Условия:

1) Ферзь может ходить по вертикали, горизонтали и наискосок.

2) Рубить ферзь может так же, как и ходить.

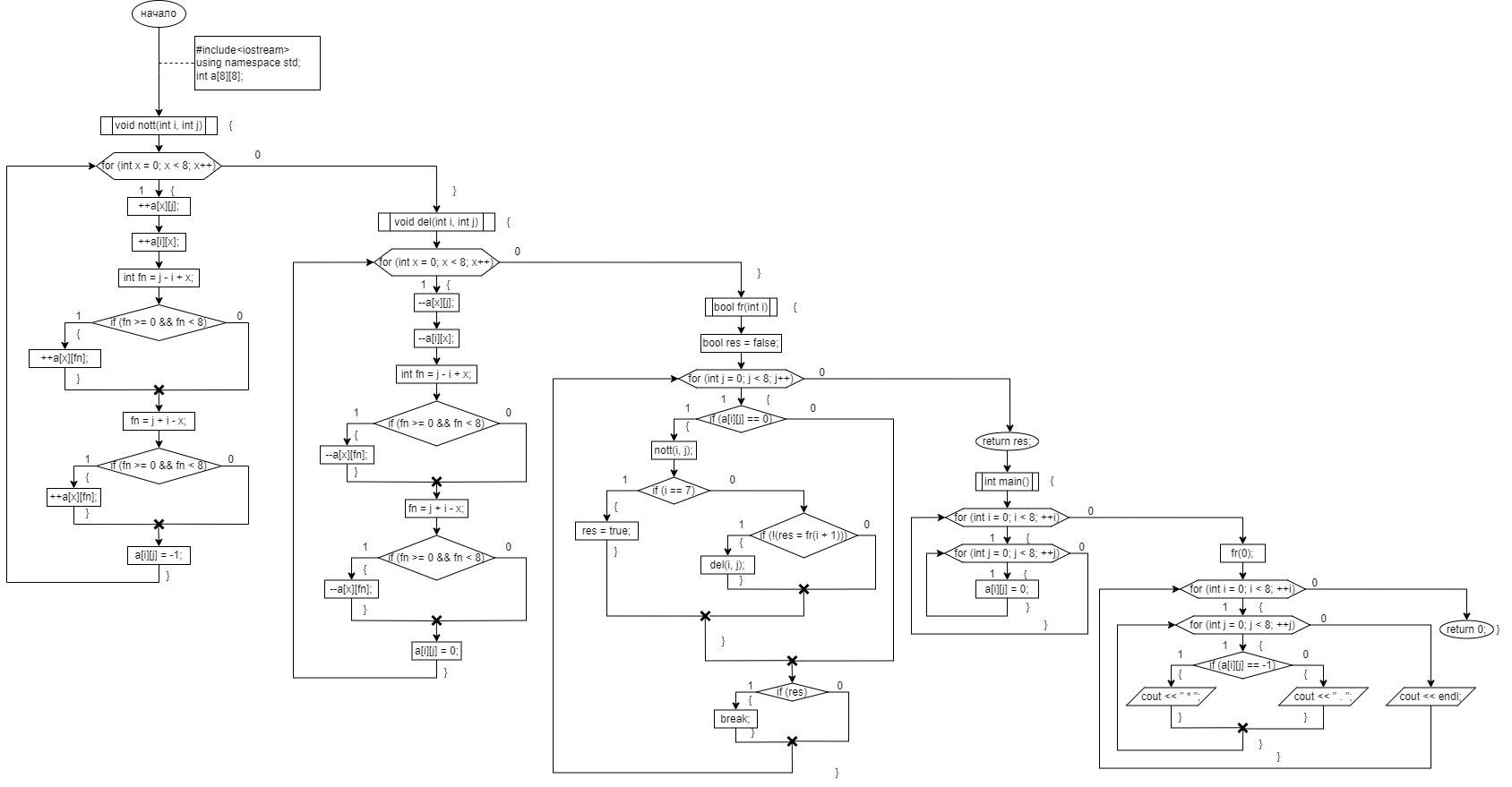
2. Анализ задачи.

1) Создание главной функции “fr”- она будет пытаться расставить ферзей. Она будет проходиться по всем столбцам i-той строки, с помощью другой функции отмечать клетки, на которые нельзя ставить ферзей. Функция будет каждый раз вызывать саму себя от следующей строки, чтобы узнать, возможно ли поставить ферзя на заданной строке. Если попытка расставить ферзей оказалась успешна- функция вернет истину; если не успешна- ложь.

2) Создание функции “nott”, которая будет заполнять доску метками, оповещающими о том, что ферзя туда ставить нельзя. С помощью цикла функция будет проходиться по всем клеткам и ставить метки по вертикали, горизонтали и диагонали. Ставить метки – добавлять к значению в ячейке 1. Если ячейка содержит значение, не равное 0 или отрицательному числу, то это количество ферзей, бьющих по этой клетке; если ячейка содержит 0- значит клетка свободна и туда можно ставить нового ферзя; если клетка содержит -1, то это значит что клетка уже занята ферзем.

3) Создание функции “del”, обратной функции “nott”, она будет удалять метки с клеток в случае неправильно поставленного ферзя(понадобиться вернуться на шаг назад). Удаление будет происходить по тому же принципу, что и при заполнении клеток метками – цикл будет проходиться по всем клеткам и удалять единицы по вертикали, горизонтали, диагонали и так же освобождать клетку, где раньше стоял ферзь(значение ячейки станет равно 0).

4) Функция “main” для начала заполнит матрицу нулями с помощью цикла for – все клетки изначально свободны. Далее будет вызвана функция “fr”, которая заполнит матрицу числами. Для вывода шахматной доски на экран понадобится еще один цикл, проходящий по матрице, который, опираясь на число, содержащееся в ячейке, будет выводить на экран либо точку – обозначение пустой клетки, либо звездочку – обозначение клетки, где стоит ферзь.

3. Блок- схема.

4. Код программы.

#include<iostream> // подключение стандартной библиотеки

using namespace std; // подключение пространства имен

int a[8][8]; // в матрице будет храниться число бьющих ферзей на каждой клетке

void nott(int i, int j) { // функция для заполнения клеток метками о том, что туда нельзя ставить ферзя

for (int x = 0; x < 8; x++) { // цикл для прохождения по всем клеткам

++a[x][j]; // в течение цикла по всей строке j добавится метка(+1 = +1 бьющий ферзь)

++a[i][x]; // в течение цикла по всему столбцу i добавляется метка(+1 = +1 бьющий ферзь)

int fn = j - i + x; // переменная для расставления меток по диагонали слева направо

if (fn >= 0 && fn < 8) { // проверка для случаев где диагональ начнется не с 1 строки

++a[x][fn]; // добавление метки по диагонали

}

fn = j + i - x; // изменение переменной для расставления меток по диагонали справо налево

if (fn >= 0 && fn < 8) { // проверка для случаев где диагональ начнется не с 1 строки

++a[x][fn]; // добавление метки по диагонали

}

a[i][j] = -1; // -1 будет обозначать то что на этой клетке стоит ферзь

}

}

void del(int i, int j) { // функция для удаления меток в случае неправильно поставленного ферзя

for (int x = 0; x < 8; x++) { // цикл для прохождения по всем клеткам

--a[x][j]; // в течение цикла по всей строке j удалится метка(-1 = -1 бьющий ферзь)

--a[i][x]; // в течение цикла по всему столбцу i удалится метка(-1 = -1 бьющий ферзь)

int fn = j - i + x; // переменная для удаления меток по диагонали слева направо

if (fn >= 0 && fn < 8) { // проверка для случаев где диагональ начнется не с 1 строки

--a[x][fn]; // удаление метки по диагонали

}

fn = j + i - x; // изменение переменной для удаления меток по диагонали справо налево

if (fn >= 0 && fn < 8) { // проверка для случаев где диагональ начнется не с 1 строки

--a[x][fn]; // удаление метки по диагонали

}

a[i][j] = 0; // обозначение, что на данную клетку сново можно ставить ферзя

}

}

bool fr(int i) { // создание функции, которая будет пытаться поставить ферзя на столбец i

bool res = false; // флажок для дальнейшей проверки, удалось ли поставить ферзя

for (int j = 0; j < 8; j++) { // прохождение по столбцам i-той строки

if (a[i][j] == 0) { // условие для проверки что ферзя никто не бьет

nott(i, j); // вызов функции для заполнения клеток метками

if (i == 7) { // проверка, чтобы узнать всех ли ферзей поставили

res = true; // если условие выполнено, значит задача решена

}

else { // действия при отрицательном ответе на условие

if (!(res = fr(i + 1))) { // проверка, будет ли попытка поставить ферзя на следующей строке неудачной

del(i, j); // функция удаления ферзя с текущей позиции

}

}

}

if (res) break; // если попытка поставить ферзя оказалось успешной, выходим из цикла

}

return res; // если попытка поставить ферзя оказалось не успешной, тогда возвратит ложь

}

int main() { // создание функции, которая будет выполнять задачу

for (int i = 0; i < 8; ++i) { // цикл для прохождения по всем строкам матрицы

for (int j = 0; j < 8; ++j) { // цикл для прохождения по всем столбцам матрицы

a[i][j] = 0; // изначально все клетки свободны(= 0)

}

}

fr(0); // вызов функции для расставления ферзей

for (int i = 0; i < 8; ++i) { // цикл для прохождения по всем строкам матрицы

for (int j = 0; j < 8; ++j) { // цикл для прохождения по всем столбцам матрицы

if (a[i][j] == -1) { // условие, чтобы указать на клетку, где стоит ферзь

cout << " \* "; // вывод положения ферзя на экран

}

else { // для клеток, в которых нет ферзя

cout << " . "; // вывод пустой клетки на экран

}

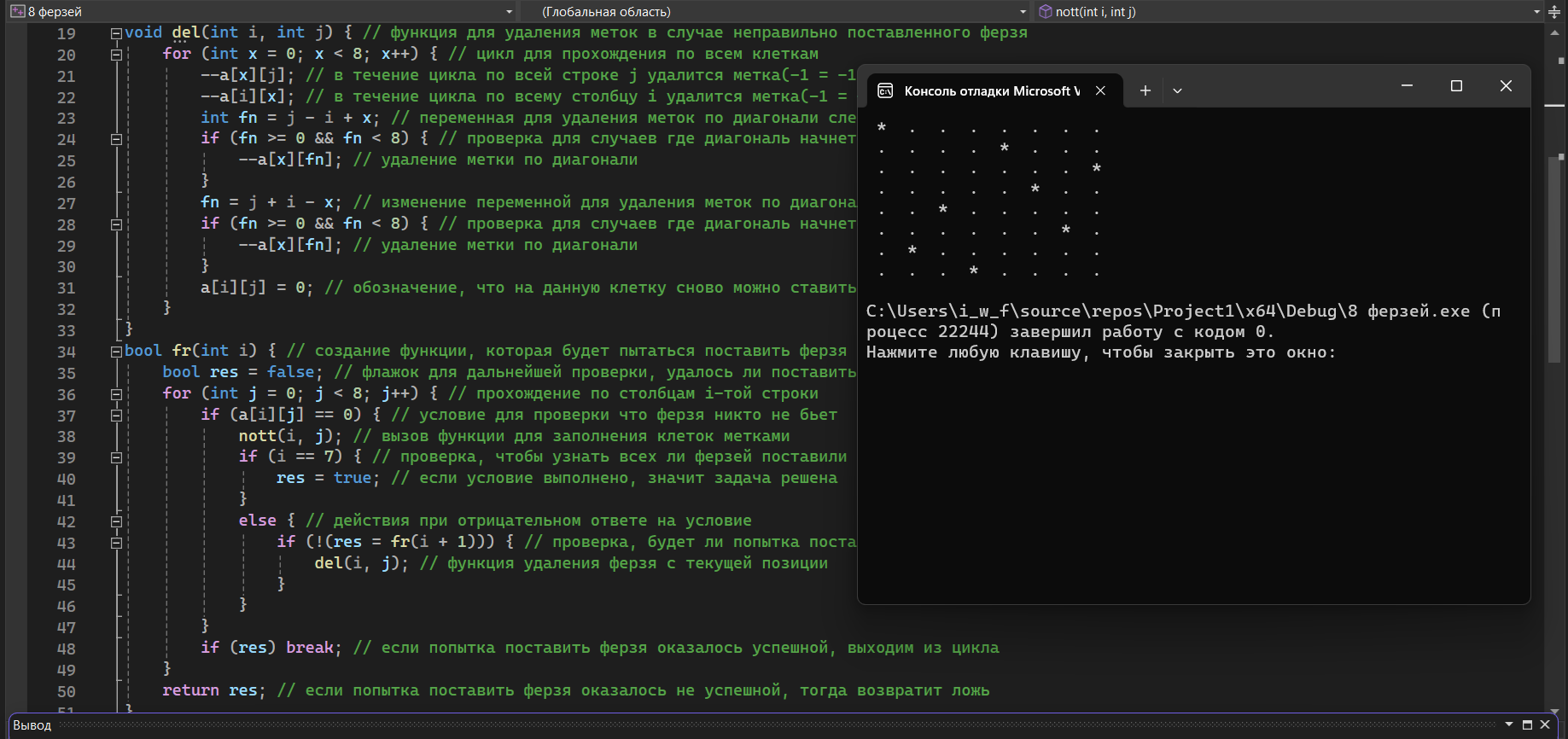
}

cout << endl; // переход на следующую строку

}

return 0; // конец работы функции

}

5. Результат работы программы.

6. Вывод: задача о 8 ферзях довольно нетривиальная, что позволяет разработчику проявить фантазию и решить индивидуальным способом и развить свои навыки.

Ссылка на работу в GitHub: <https://github.com/slurree/first.git>