Бюджетное учреждение высшего образования

Ханты-Мансийского автономного округа

«Сургутский государственный университет»

Политехнический институт

Кафедра автоматики и компьютерных систем

**Отчет**

по лабораторной работе № 2 «Разветвляющиеся алгоритмы»

по дисциплине «Программирование и основы алгоритмизации»

Выполнил: Скорба Р.В.

студент группы 609-11

Проверил: Гришмановский П. В.

доцент кафедры автоматики и

компьютерных систем

Сургут

2021 г.

**Цель работы:**

Закрепление теоретических знаний конструкции выбора структурного программирования, получение практических навыков разработки разветвляющихся алгоритмов, построение логических выражений и использование условных операторов.

**Задание:**

Найти общее решение вычислительной задачи в соответствии с индивидуальным заданием, выданным преподавателем, разработать алгоритм решения и реализовать программу.

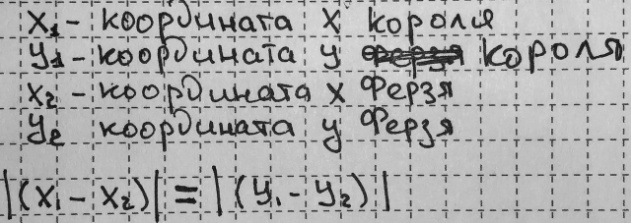
**Вариант №23:** Заданы координаты черного короля, белых ферзя и ладьи на шахматной доске. Определить имеет ли место мат.

**Вариант 23.**

**Формальное описание задачи.**

Пользователь вводит поочередно координаты X, Y для: короля, ферзя и ладьи. После чего программа просчитывает, бьет ли одна из фигур короля, если бьет - король пытается сбежать, если нет - мат не поставлен.

Считается, что ферзь бьет короля, если ферзь может сходить так, что координаты ферзя и короля будут равны. Ферзь может ходить по вертикали, горизонтали и диагонали, на любое расстояние. Чтобы узнать, находится ли король под ударом, по горизонтальной оси, необходимо сравнить координату X ферзя и короля, если равны, то значит ферзь бьет короля; для вертикальной оси необходимо сравнить координаты Y; а для диагонали, необходимо воспользоваться формулой (если условие, которое представлено в формуле верное, то ферзь бьет короля по диагонали):



Ладья может ходить на любое расстояние по вертикали и горизонтали. Чтобы узнать, находится ли король под ударом ладьи, по горизонтальной оси, необходимо сравнить координату X ладьи и короля; для вертикальной оси необходимо сравнить координаты Y.

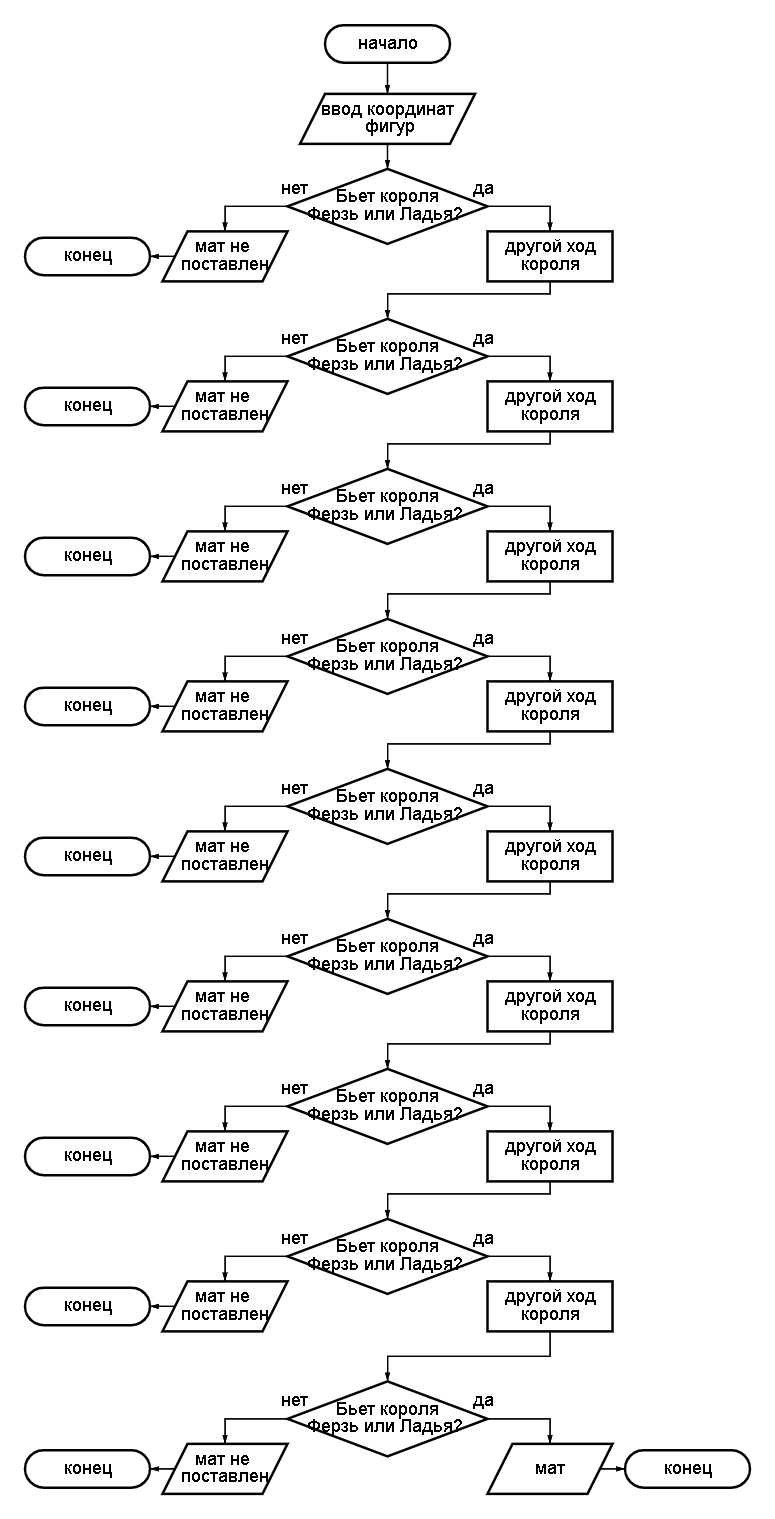
Король может передвигаться в любую сторону, но на одну клетку, соответственно нужно проверить 1 начальное положение короля и 8 возможных вариантов «побега». При попытке сбежать проверяются каждые новые координаты короля на то, бьет ли его одна из фигур или нет, если хоть на одном из возможных ходов фигуру короля не бьет, соответственно мат не поставлен и программа выводит ответ, “Мат не поставлен”.

**Алгоритм программы.**

Для решения данной задачи необходимо:

1. Ввести координаты для: короля, ферзя, ладьи.
2. Просчитать бьет ли хоть одна из фигур короля.
3. Если бьет, то проверить может ли король сбежать.
4. Если король убегает, программа выводит, “Мат не поставлен!”, в противном случае программа выводит, “Мат поставлен!”.

**Блок-схема алгоритма программы.**

****

**Листинг программы.**

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main () {

system ("chcp 1251 > nul");

int kx, kx1, ky, ky1, rx, ry, qx, qy;

printf("Введите значение координат Короля от 1-8: ");

scanf("%d%d", &kx, &ky);

printf("Введите значение координат Ладьи от 1-8: ");

scanf("%d%d", &rx, &ry);

printf("Введите значение координат Ферзя от 1-8: ");

scanf("%d%d", &qx, &qy);

kx1 = kx;

ky1 = ky;

if ((rx == kx) || (qx == kx) || (ry == ky) || (qy == ky) || abs(qx - kx) == abs(qy - ky)) {

kx1 = kx + 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((kx1 == 9) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

kx1 = kx - 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((kx1 == 0) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

kx1 = kx + 1;

ky1 = ky + 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((kx1 == 9) || (ky1 == 9) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

kx1 = kx + 1;

ky1 = ky - 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((kx1 == 9) || (ky1 == 0) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

ky1 = ky + 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((ky1 == 9) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

ky1 = ky - 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((ky1 == 0) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

kx1 = kx - 1;

ky1 = ky - 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((kx1 == 0) || (ky1 == 0) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1)) {

kx1 = kx - 1;

ky1 = ky + 1;

}

else {

printf("Мат не поставлен!");

return 0;

}

if ((kx1 == 0) || (ky1 == 9) || (rx == kx1) || (qx == kx1) || (ry == ky1) || (qy == ky1) || abs(qx - kx1) == abs(qy - ky1))

{

kx1 = kx;

ky1 = ky;

printf("Мат!\n");

}

printf("kx = %d\nky = %d\n", kx1, ky1);

system("pause");

return 0;

}

**Пояснения к программе.**

Подключены 3 библиотеки: <stdio.h> - для функций printf & scanf, <stdlib.h> - для использования функции system(), <math.h> - для математических вычислений.

Для того чтобы пользователь понимал что от него требуется, используется ‘printf();’, в котором описано условие.

Чтобы программа лишний раз не проверяла каждое условие, в случае если короля не бьет другая фигура, программа сразу выводит ответ и завершает свою работу с помощью return 0;

**Вывод:** В задаче для всех переменных используется тип integer, так как все величины после всех вычислений должны быть целочисленные. В программе есть оператор выбора «if» и «else», так же в программе используются логические выражения, для проверки нескольких условий одновременно.