МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ СИСТЕМ И ТЕХНОЛОГИЙ

Кафедра «Информационные системы»

Дисциплина «Основы алгоритмизации и программирования»

Лабораторная работа №6

Работа с клавиатурой

Выполнил:

студент гр. ПИбд-12

Дозорова Алена

Ульяновск, 2021 г

Задание лабораторной работы

1. Собрать код игры из исходников
   * Поиграть в игру 2-3 минуты
2. Переделать лабораторную работу №4 добавив в неё возможность выбирать отображаемый рисунок клавишами VK\_LEFT (к предыдущему рисунку) и VK\_RIGHT (к следующему рисунку).
3. Переделать лабораторную работу №4 добавив в неё возможность выбирать размер отображаемого рисунка клавишами VK\_DOWN (уменьшить размер рисунка) и VK\_UP (увеличить размер рисунка) .
4. Добавить к коду изображения из лабораторной работы №5.
5. Придумать набор клавиш для переключения между изображениями из Лабораторных работ №4 и №5. Собрать в единой программе (Лабораторная работа №6) ВСЕ рисунки, выполненные на данный момент в классе и дома.
6. Переключение между рисунками обеспечить при помощи подходящих клавиш. Конкретные клавиши выбрать самостоятельно.
7. Нарисовать блоксхемы для всех Switch, найденных в коде программы
8. Реализовать это переключение.

Код программы (switch case)

switch (number\_task)

{

case 1:

game(hdc);

break;

case 2:

RecursiveCross(hdc, 200, 160, size, RGB(r, 0, 0));

break;

case 3:

RecursiveTriangle(hdc, 200, 160, size);

break;

case 4:

RecursiveCircle(hdc, 200, 160, size);

break;

case 5:

pic(hdc);

break;

case 6:

cicle(hdc);

break;

case 7:

StClausAuto(hdc, 10, 10);

break;

case 8:

StClausAuto(hdc, 10, 10);

break;

case 9:

Snowman(hdc, 10, 10);

break;

case 10:

Logo(hdc, 10, 10);

break;

case 11:

RecursiveImage1\_1(hdc, 200, 160, 80);

break;

case 12:

RecursiveImage1\_2(hdc, 550, 150, 80);

RecursiveImage1\_3(hdc, 800, 150, 80);

RecursiveImage1\_4(hdc, 80, 400, 80);

RecursiveImage1\_5(hdc, 260, 400, 80);

RecursiveImage1\_6(hdc, 550, 400, 80);

RecursiveImage1\_7(hdc, 800, 500, 80);

break;

case 13:

RecursiveImage2\_1(hdc, 200, 100, 80);

RecursiveImage2\_2(hdc, 400, 150, 60);

RecursiveImage2\_3(hdc, 600, 150, 60);

RecursiveImage2\_4(hdc, 800, 100, 80);

break;

case 14:

RecursiveImage2\_5(hdc, 200, 400, 80);

RecursiveImage2\_6(hdc, 600, 400, 80);

RecursiveImage2\_7(hdc, 800, 400, 80);

RecursiveImage2\_8(hdc, 1000, 400, 80);

break;

case 15:

RecursiveImage3\_1(hdc, 100, 100, 60);

RecursiveImage3\_2(hdc, 500, 200, 60);

RecursiveImage3\_3(hdc, 800, 200, 60);

RecursiveImage3\_4(hdc, 100, 400, 60);

RecursiveImage3\_5(hdc, 500, 400, 60);

break;

case 16:

RecursiveImage4\_1(hdc, 100, 100, 80);

RecursiveImage4\_2(hdc, 450, 100, 80);

RecursiveImage4\_3(hdc, 800, 100, 80);

RecursiveImage4\_4(hdc, 150, 300, 80);

break;

case 17:

RecursiveImage5\_1(hdc, 200, 200, 80);

RecursiveImage5\_1(hdc, 500, 100, 20);

break;

case 18:

RecursiveImage6\_1(hdc, 200, 200, 60);

break;

case 19:

RecursiveImage7\_1(hdc, 500, 200, 60);

break;

case 20:

RecursiveImage9\_1(hdc, 800, 200, 60);

break;

}

switch (wParam)

{

case VK\_DOWN:

moveDown();

size -= 10;

InvalidateRect(hWnd, NULL, TRUE);

break;

case VK\_LEFT:

r -= 10;

moveToLeft();

InvalidateRect(hWnd, NULL, TRUE);

break;

case VK\_UP:

moveUp();

size += 10;

InvalidateRect(hWnd, NULL, TRUE);

break;

case VK\_RIGHT:

r -= 10;

moveToRight();

InvalidateRect(hWnd, NULL, TRUE);

break;

case VK\_RETURN:

if (insideGoal(aimX, aimY)) {

hit++;

}

else {

missed++;

}

InvalidateRect(hWnd, NULL, TRUE);

break;

case VK\_F1:

number\_task -= 1;

InvalidateRect(hWnd, NULL, TRUE);

break;

case VK\_F2:

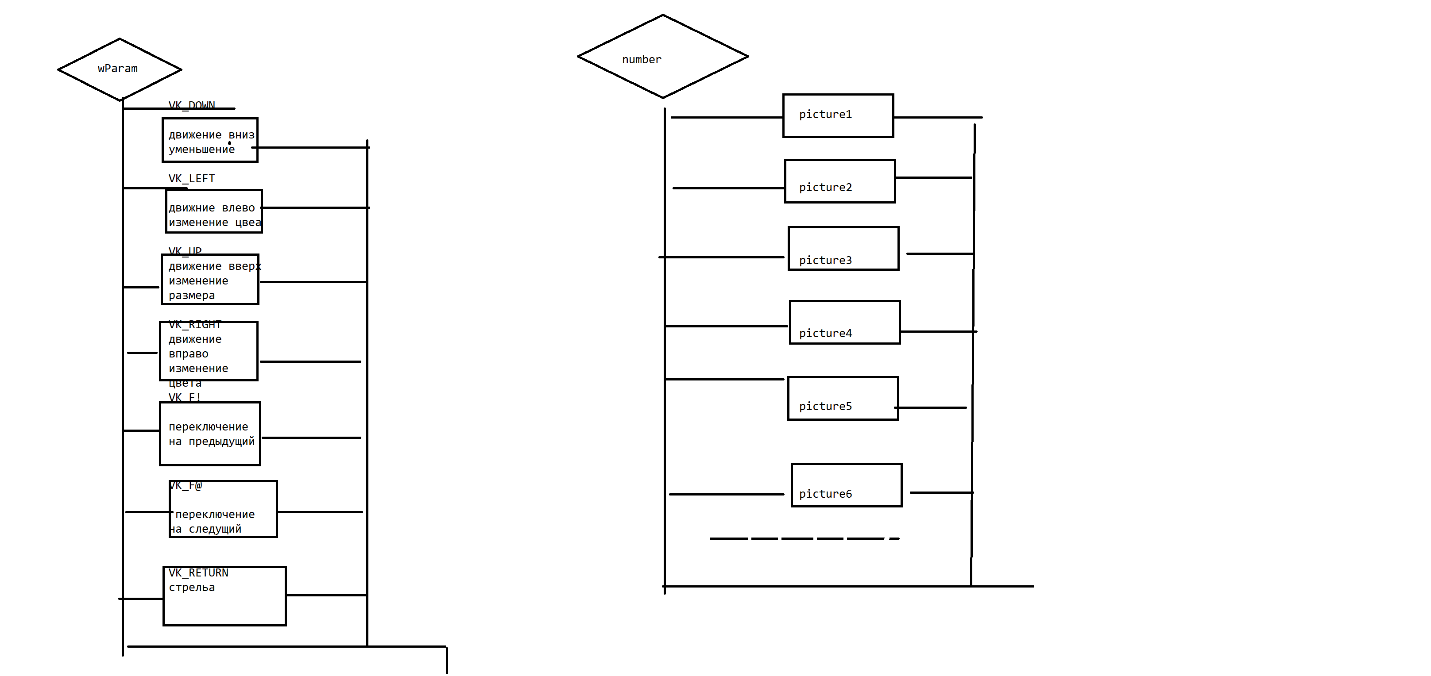
number\_task += 1;

InvalidateRect(hWnd, NULL, TRUE);

break;

}

Блок-схемы



Заключение

В ходе этой лабораторной я оптимизировала сдачу заданий лабораторной работы, используя изученные конструкцию switch case и коды клавиш клавиатуры.