ПОЯСНЕНИЕ КОДА

Практическая работа №6. Мини-игра "Камень, Ножницы, Бумага" с компьютером

**Описание программы:**

Данная программа представляет собой простую игру «Камень, ножницы, бумага» с графическим интерфейсом на основе библиотеки **Raylib**. Игрок может выбирать один из трех вариантов с помощью клавиш **R, P, S** или кликая на кнопки. Компьютер делает случайный выбор, после чего отображается результат игры.

**Разбор кода:**

#include "raylib.h" – **подключение библиотеки Raylib для работы с графикой**

#include <stdlib.h> - **используются для генерации числа**

#include <time.h>

typedef enum { ROCK, PAPER, SCISSORS, NONE } Choice; - **перечисление для выбора игрока/компьютер**

Choice GetComputerChoice() {

return (Choice)(rand() % 3);

}

**Функция для получения случайного выбора компьютера**

const char\* GetResult(Choice player, Choice computer, Color\* textColor) {

if (player == computer) {

\*textColor = ORANGE;

return "Draw!";

}

if ((player == ROCK && computer == SCISSORS) ||

(player == PAPER && computer == ROCK) ||

(player == SCISSORS && computer == PAPER)) {

\*textColor = GREEN;

return "You Win!";

}

\*textColor = RED;

return "You Lose!";

}

**Определяет кто выиграл, поиграли, или произошла ничья**

**Победа - зеленый,проигрыш красный, ничья оранжевый**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int main() {

int screenWidth = 1000;

int screenHeight = 600;

InitWindow(screenWidth, screenHeight, "Rock, Paper, Scissors - Sagantay Adil CS-104(s)");

SetTargetFPS(60);

srand(static\_cast<unsigned int>(time(NULL)));

int screenWidth = 1000; **- разрешение экрана 1000х600**

int screenHeight = 600;

InitWindow(screenWidth, screenHeight, "Rock, Paper, Scissors - Sagantay Adil CS-104(s)"); - **создание окна и его заголовок**

SetTargetFPS(60); - **установка 60 FPS для плавного отображения**

Texture2D rockSprite = LoadTexture("resources/rock.png");

Texture2D paperSprite = LoadTexture("resources/paper.png");

Texture2D scissorsSprite = LoadTexture("resources/scissors.png");

**Загружаются изображения для камня, бумаги и ножницы**

Choice playerChoice = NONE; - **хранят выбор игрока**

Choice computerChoice = NONE; - **хранят выборка компьютера**

bool gameStarted = false; - **проверяет началась ли игра**

bool selectionMade = false; - **следит за тем был уже ли сделан выбор**

Color resultColor = BLACK; - **хранит цвет результата**

Rectangle rockButton = { screenWidth / 2 - 225, 200, 100, 50 };

Rectangle paperButton = { screenWidth / 2 - 50, 200, 100, 50 };

Rectangle scissorsButton = { screenWidth / 2 + 125, 200, 120, 50 };

Rectangle resetButton = { screenWidth / 2 - 75, 300, 150, 50 };

**Определение кнопок для выбора камня, бумаги, ножницы и кнопки сброса**

float cornerRadius = 0.4f; - **радиус округления кнопок**

while (!WindowShouldClose()) { - **главный цикл игры выполняющий до закрытия игр**

Vector2 mousePoint = GetMousePosition();

bool hoveredRock = CheckCollisionPointRec(mousePoint, rockButton);

bool hoveredPaper = CheckCollisionPointRec(mousePoint, paperButton);

bool hoveredScissors = CheckCollisionPointRec(mousePoint, scissorsButton);

bool hoveredReset = CheckCollisionPointRec(mousePoint, resetButton);

**получает позицию мыши и проверяет наведена ли мышь на кнопки**

if (IsKeyPressed(KEY\_SPACE) || (IsMouseButtonPressed(MOUSE\_LEFT\_BUTTON) && hoveredReset)) {

playerChoice = NONE;

computerChoice = NONE;

gameStarted = false;

selectionMade = false;

resultColor = BLACK;

}

**Если нажата пробел или кнопка сброса, то игра начинается заново**

if (!gameStarted && !selectionMade) {

if (IsKeyPressed(KEY\_R)) playerChoice = ROCK;

if (IsKeyPressed(KEY\_P)) playerChoice = PAPER;

if (IsKeyPressed(KEY\_S)) playerChoice = SCISSORS;

if (playerChoice != NONE) {

computerChoice = GetComputerChoice();

gameStarted = true;

selectionMade = true;

}

}

**Выбор игрока с клавиатуры. Игрок может выбрать вариант клавиши R, P, S.**

if (!selectionMade && IsMouseButtonPressed(MOUSE\_LEFT\_BUTTON)) {

if (hoveredRock) playerChoice = ROCK;

if (hoveredPaper) playerChoice = PAPER;

if (hoveredScissors) playerChoice = SCISSORS;

if (playerChoice != NONE) {

computerChoice = GetComputerChoice();

gameStarted = true;

selectionMade = true;

}

}

**Выбор игрока мышкой, если игрок кликнул по кнопке, то выбор фиксируется**

BeginDrawing();

ClearBackground(GRAY);

**Очищает экран и устанавливает фон серого цвета**

DrawText("Press R - Rock, P - Paper, S - Scissors to play or click buttons", screenWidth / 2 - 300, 150, 20, BLACK);

DrawText("Press SPACE or click reset to restart", screenWidth / 2 - 200, 265, 20, BLACK);

**Отображение текста**

Color buttonColor = LIGHTGRAY;

Color hoverColor = DARKGRAY;

**Кнопка до наведения мыши – светло серый, если наведить мышь на кнопку – темно серый**

DrawRectangleRounded(rockButton, cornerRadius, 6, hoveredRock ? hoverColor : buttonColor);

DrawRectangleRounded(paperButton, cornerRadius, 6, hoveredPaper ? hoverColor : buttonColor);

DrawRectangleRounded(scissorsButton, cornerRadius, 6, hoveredScissors ? hoverColor : buttonColor);

DrawRectangleRounded(resetButton, cornerRadius, 6, hoveredReset ? hoverColor : buttonColor);

**Рисование кнопки с закругленными углами**

DrawText("Rock", rockButton.x + 25, rockButton.y + 15, 20, BLACK);

DrawText("Paper", paperButton.x + 25, paperButton.y + 15, 20, BLACK);

DrawText("Scissors", scissorsButton.x + 15, scissorsButton.y + 15, 20, BLACK);

DrawText("Reset", resetButton.x + 40, resetButton.y + 15, 20, BLACK);

**Отображение кнопки по X/Y**

if (gameStarted) {

DrawText("Your choice:", screenWidth / 2 - 200, 350, 20, BLACK);

DrawText("Computer choice:", screenWidth / 2 + 50, 350, 20, BLACK);

Vector2 playerPos = { screenWidth / 2 - 200, 380 };

Vector2 computerPos = { screenWidth / 2 + 50, 380 };

if (playerChoice == ROCK)

DrawTexturePro(rockSprite, { 0, 0, (float)rockSprite.width, (float)rockSprite.height }, { playerPos.x, playerPos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (playerChoice == PAPER)

DrawTexturePro(paperSprite, { 0, 0, (float)paperSprite.width, (float)paperSprite.height }, { playerPos.x, playerPos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (playerChoice == SCISSORS)

DrawTexturePro(scissorsSprite, { 0, 0, (float)scissorsSprite.width, (float)scissorsSprite.height }, { playerPos.x, playerPos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (computerChoice == ROCK)

DrawTexturePro(rockSprite, { 0, 0, (float)rockSprite.width, (float)rockSprite.height }, { computerPos.x, computerPos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (computerChoice == PAPER)

DrawTexturePro(paperSprite, { 0, 0, (float)paperSprite.width, (float)paperSprite.height }, { computerPos.x, computerPos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (computerChoice == SCISSORS)

DrawTexturePro(scissorsSprite, { 0, 0, (float)scissorsSprite.width, (float)scissorsSprite.height }, { computerPos.x, computerPos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

DrawText(GetResult(playerChoice, computerChoice, &resultColor), screenWidth / 2 - 50, 500, 30, resultColor);

}

EndDrawing();

}

**Выводит результат игры. Выводит Что игрок выбрал и компьютер , показывая изображения.**

UnloadTexture(rockSprite);

UnloadTexture(paperSprite);

UnloadTexture(scissorsSprite);

CloseWindow();

return 0;

**Освобождает загруженные текстуры и закрывает окно**

Мини-игра "Камень, Ножницы, Бумага" в режиме двух игроков

**Описание программы:**

Эта программа представляет собой консольную игру "Камень, ножницы, бумага" для двух игроков с графическим интерфейсом, созданным с помощью библиотеки Raylib.

**Разбор кода:**

#include "raylib.h"

#include <stdlib.h>

#include <time.h>

typedef enum { ROCK, PAPER, SCISSORS, NONE } Choice;

const char\* DetermineWinner(Choice player1, Choice player2, Color\* textColor) {

if (player1 == player2) {

\*textColor = ORANGE;

return "Draw!";

}

if ((player1 == ROCK && player2 == SCISSORS) ||

(player1 == PAPER && player2 == ROCK) ||

(player1 == SCISSORS && player2 == PAPER)) {

\*textColor = GREEN;

return "Player 1 Wins!";

}

\*textColor = RED;

return "Player 2 Wins!";

}

**Функция определяет победителя , сравнивая выборы двух игроков**

int main() {

int screenWidth = 1000;

int screenHeight = 600;

InitWindow(screenWidth, screenHeight, "Rock, Paper, Scissors - Two Players");

SetTargetFPS(60);

Разрешение экрана 1000х600, 60 FPS, Заголовок окна

Texture2D rockSprite = LoadTexture("resources/rock.png");

Texture2D paperSprite = LoadTexture("resources/paper.png");

Texture2D scissorsSprite = LoadTexture("resources/scissors.png");

**Загружают изображения**

Choice player1Choice = NONE; - **хранят выбор первого игрока**

Choice player2Choice = NONE; - **хранят выбор второго игрока**

bool gameStarted = false;  **- проверяет началаись ли игра**

bool player1MadeChoice = false; - **следит был ли выбран выбор первого игрока**

bool player2MadeChoice = false; - **следит был ли выбран выбор второго игрока**

Color resultColor = BLACK; - **хранит цвет результата**

Rectangle resetButton = { screenWidth / 2 - 75, 300, 150, 50 };

float cornerRadius = 0.4f;

**Координты и размеры кнопки, округление радиуса кнопки**

while (!WindowShouldClose()) {

Vector2 mousePoint = GetMousePosition();

bool hoveredReset = CheckCollisionPointRec(mousePoint, resetButton);

if (IsKeyPressed(KEY\_SPACE) || (IsMouseButtonPressed(MOUSE\_LEFT\_BUTTON) && hoveredReset)) {

player1Choice = NONE;

player2Choice = NONE;

gameStarted = false;

player1MadeChoice = false;

player2MadeChoice = false;

resultColor = BLACK;

}

**Получает текущая позиция мышь, проверяет наведена ли мышь на кнопку Reset , если нажата**

**SPACE или мышь кликнута по кнопке, игра начинается заново**

if (!gameStarted) {

if (!player1MadeChoice) {

if (IsKeyPressed(KEY\_R)) { player1Choice = ROCK; player1MadeChoice = true; }

if (IsKeyPressed(KEY\_P)) { player1Choice = PAPER; player1MadeChoice = true; }

if (IsKeyPressed(KEY\_S)) { player1Choice = SCISSORS; player1MadeChoice = true; }

}

if (player1MadeChoice && !player2MadeChoice) {

if (IsKeyPressed(KEY\_A)) { player2Choice = ROCK; player2MadeChoice = true; }

if (IsKeyPressed(KEY\_W)) { player2Choice = PAPER; player2MadeChoice = true; }

if (IsKeyPressed(KEY\_D)) { player2Choice = SCISSORS; player2MadeChoice = true; }

}

if (player1MadeChoice && player2MadeChoice) {

gameStarted = true;

}

}

**Игроки нажимают клавиши**

**Игрок 1 – R, P, S**

**Игрок 2 – A, W, D**

**Как только оба игрока сделали выбора, игра начинается gameStarted = true**

BeginDrawing();

ClearBackground(GRAY);

DrawText("Player 1: Press R - Rock, P - Paper, S - Scissors", 50, 150, 20, BLACK);

DrawText("Player 2: Press A - Rock, W - Paper, D - Scissors", 50, 180, 20, BLACK);

DrawText("Press SPACE or click reset to restart", 50, 210, 20, BLACK);

if (!player1MadeChoice) {

DrawText("Player 1, make your choice!", 50, 250, 20, RED);

}

else if (!gameStarted) {

DrawText("Player 1 has chosen!", 50, 250, 20, GREEN);

}

if (player1MadeChoice && !player2MadeChoice) {

DrawText("Player 2, make your choice!", 50, 280, 20, RED);

}

if (player1MadeChoice && player2MadeChoice) {

DrawText("Both players have chosen!", 50, 310, 20, GREEN);

}

**Отображение подсказок для игроков**

**Если Игрок 1 еще не сделал выбор, отображается красный цвет** Player 1, make your choice!

**Если Игрок 1 выбрал, а Игрок 2 еще не выбрал, то выводится зеленый** Player 1 has chosen!

**Если Игрок 1 выбрал, а Игрок 2 нет, то выводится красный** Player 2, make your choice!

**Если обо игрока сделали выбор то выводится зеленый** Both players have chosen!

DrawRectangleRounded(resetButton, cornerRadius, 6, hoveredReset ? DARKGRAY : LIGHTGRAY);

DrawText("Reset", resetButton.x + 40, resetButton.y + 15, 20, BLACK);

**Рисует закругленный прямоугольник**

**Отображается кнопки Reset по координату X/Y**

if (gameStarted) {

DrawText("Player 1's choice:", 200, 350, 20, BLACK);

DrawText("Player 2's choice:", 600, 350, 20, BLACK);

Vector2 player1Pos = { 200, 380 };

Vector2 player2Pos = { 600, 380 };

if (player1Choice == ROCK)

DrawTexturePro(rockSprite, { 0, 0, (float)rockSprite.width, (float)rockSprite.height }, { player1Pos.x, player1Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player1Choice == PAPER)

DrawTexturePro(paperSprite, { 0, 0, (float)paperSprite.width, (float)paperSprite.height }, { player1Pos.x, player1Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player1Choice == SCISSORS)

DrawTexturePro(scissorsSprite, { 0, 0, (float)scissorsSprite.width, (float)scissorsSprite.height }, { player1Pos.x, player1Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player2Choice == ROCK)

DrawTexturePro(rockSprite, { 0, 0, (float)rockSprite.width, (float)rockSprite.height }, { player2Pos.x, player2Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player2Choice == PAPER)

DrawTexturePro(paperSprite, { 0, 0, (float)paperSprite.width, (float)paperSprite.height }, { player2Pos.x, player2Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player2Choice == SCISSORS)

DrawTexturePro(scissorsSprite, { 0, 0, (float)scissorsSprite.width, (float)scissorsSprite.height }, { player2Pos.x, player2Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

DrawText(DetermineWinner(player1Choice, player2Choice, &resultColor), screenWidth / 2 - 50, 500, 30, resultColor);

}

DrawText("Player 1's choice:", 200, 350, 20, BLACK);

DrawText("Player 2's choice:", 600, 350, 20, BLACK);

**Выводит информацию о выборе игроков**

Vector2 player1Pos = { 200, 380 };

Vector2 player2Pos = { 600, 380 };

**Указание позиции для изображений**

if (player1Choice == ROCK)

DrawTexturePro(rockSprite, { 0, 0, (float)rockSprite.width, (float)rockSprite.height }, { player1Pos.x, player1Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player1Choice == PAPER)

DrawTexturePro(paperSprite, { 0, 0, (float)paperSprite.width, (float)paperSprite.height }, { player1Pos.x, player1Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player1Choice == SCISSORS)

DrawTexturePro(scissorsSprite, { 0, 0, (float)scissorsSprite.width, (float)scissorsSprite.height }, { player1Pos.x, player1Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

**Отображение картинок выбора игрока 1**

if (player2Choice == ROCK)

DrawTexturePro(rockSprite, { 0, 0, (float)rockSprite.width, (float)rockSprite.height }, { player2Pos.x, player2Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player2Choice == PAPER)

DrawTexturePro(paperSprite, { 0, 0, (float)paperSprite.width, (float)paperSprite.height }, { player2Pos.x, player2Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

if (player2Choice == SCISSORS)

DrawTexturePro(scissorsSprite, { 0, 0, (float)scissorsSprite.width, (float)scissorsSprite.height }, { player2Pos.x, player2Pos.y, 100, 100 }, { 0, 0 }, 0, WHITE);

**Отображение картинок выбора игрока 2**

**В зависимости от выбора игрока, загружается соответствующая картинка**

**Исходные размеры берутся из загруженных**

**Изображение масштабируется 100х100**

DrawText(DetermineWinner(player1Choice, player2Choice, &resultColor), screenWidth / 2 - 50, 500, 30, resultColor);

**Определение победителя и вывод результата**

EndDrawing();

}

UnloadTexture(rockSprite);

UnloadTexture(paperSprite);

UnloadTexture(scissorsSprite);

CloseWindow();

return 0;

}