МИНИСТЕРСТВО ВЫСШЕГО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное бюджетное образовательное учреждение**

**науки и высшего образования**

**К Г Э У «КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»**

# (ФГБОУ ВО «КГЭУ»)

**Кафедра Информатики и информационных управляющих систем**

**ДОКЛАД № 1**

**дисциплина «Алгоритмизация и программирование»**

|  |  |
| --- | --- |
| **Выполнили:** | Соловьёв Леонид,  Матижева Анна |
| **Группа:** | ПИ-1-22 |
| **Проверила:** | Шорина Т.В. |
|  |  |

Казань, 2023

**Написание простейшей игры “Арканоид” с использованием библиотеки pygame**

**Создание окна с фоновой картинкой**

Для того, чтобы начать работу с библиотекой необходимо ее установить с помощью pip (package-management system как apt или pacman)

После установки, начинаем писать код. C помощью команды pygame.init() запускаем pygame. Создадим переменные WIDTH(1200) и HEIGHT(800), которые будут означать размер окна. И resolution = (WIDTH, HEIGHT)

Само окно создаётся c помощью переменной screen и функции pygame.display.set\_mode(resolution)

При запуске этого кода, создастся окно с написанным разрешением.  
Но оно не отлавливает события, а значит и прервать данную программу мы не сможем. Чтобы это сделать, необходимо создать бесконечный цикл(while True), в котором будут отлавливаться события(for event in pygame.event.get()), обновляться состояние и отображаться состояние(pygame.display.flip()).

Создаем бесконечный цикл while True, добавляем цикл for event in pygame.event.get(), который отвечает за отлов событий. Добавим условие выхода из программы(if event.type == pygame.QUIT), действия pygame.display.quit()(закрытие окна) и exit()(остановка программы).

C добавленным циклом мы можем беспрепятственно выходить из программы в любой момент.

Перед циклом создадим переменную background\_image = pygame.image.load(имя файла с картинкой в кавычках), а также background\_position = (0, 0)

Добавим в while строку screen.blit(background\_image, background\_position), которая отображает на screen background\_image в background\_position.

При запуске этого кода, создастся окно c фоновой картинкой.

**Создание платформы**

Создадим переменные paddle\_width(200), paddle\_height(20), start\_width(WIDTH // 2 – paddle\_width // 2), space\_paddle\_screen(4), start\_height(HEIGHT – paddle\_height – space\_paddle\_screen), paddle и color\_of\_paddle.

paddle = pygame.Rect(start\_width, start\_height, paddle\_width, paddle\_height)

color\_of\_paddle = pygame.Color(‘purple’)

Класс Rect создаёт прямоугольник в окне на позиции (start\_width, start\_height), размер прямоугольника (paddle\_width, paddle\_height)

Для того, чтобы отобразить этот прямоугольник необходимо ПОСЛЕ screen.blit(…) написать команду pygame.draw.rect(screen, color\_of\_paddle, paddle)

pygame.draw.rect(…) отображает прямоугольник paddle цвета color\_of\_paddle на screen.

Отображать платформу надо после заднего фона, иначе он перекроет платформу и мы её не увидим.

Для того, чтобы управлять платформой необходимо в основной цикл добавить   
снятие маски нажатых клавиш(keys = pygame.key.get\_pressed())

Добавим условия на нажатые клавиши keys[pygame.K\_LEFT], keys[pygame.K\_RIGHT], keys[pygame.K\_a] и keys[pygame.K\_d] + для того, чтобы платформа не выходила за пределы окна добавим условия на выход правого и левого края прямоугольника за границы. Если нажата стрелка влево или A, то значение левого края прямоугольника уменьшается на значение скорости платформы, для стрелки вправо и d соответственно – увеличивается значение правого края на значение скорости:

if (keys[pygame.K\_LEFT] or keys[pygame.K\_a]) and paddle.left > 0:  
 paddle.left -= paddle\_speed  
elif (keys[pygame.K\_RIGHT] or keys[pygame.K\_d]) and paddle.right < WIDTH:  
 paddle.right += paddle\_speed

# current code

import pygame

# start pygame

pygame.init()

# resolution of window

WIDTH, HEIGHT = 1200, 800

resolution = (WIDTH, HEIGHT)

#create window

screen = pygame.display.set\_mode(resolution)

# background settings

background\_image = pygame.image.load('background.jpg')

background\_position = (0, 0)

# paddle settings

paddle\_width = 200

paddle\_height = 20

paddle\_speed = 1

color\_of\_paddle = pygame.Color('purple')

# paddle position

start\_width = WIDTH // 2 - paddle\_width // 2

space\_paddle\_screen = 4

start\_height = HEIGHT - paddle\_height - space\_paddle\_screen

# create paddle

paddle = pygame.Rect(start\_width, start\_height, paddle\_width, paddle\_height)

# main loop

while True:

for event in pygame.event.get():

# quit and stop programm by cross

if event.type == pygame.QUIT:

pygame.display.quit()

exit()

# place background on the screen

screen.blit(background\_image, background\_position)

# draw paddle

pygame.draw.rect(screen, color\_of\_paddle, paddle)

# current pressed keys

keys = pygame.key.get\_pressed()

# control of paddle

if (keys[pygame.K\_LEFT] or keys[pygame.K\_a]) and paddle.left > 0:

paddle.left -= paddle\_speed

elif (keys[pygame.K\_RIGHT] or keys[pygame.K\_d]) and paddle.right < WIDTH:

paddle.right += paddle\_speed

#update screen

pygame.display.flip()