

**Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
"Уфимский государственный авиационный технический университет"**

Кафедра Высокопроизводительных вычислительных технологий и систем

Дисциплина: Программирование

Отчет по лабораторной работе № _5_

Тема: «Использование сторонних библиотек. Работа с готовыми классами в C++»

Группа ПМ-153	Фамилия И.О.	Подпись	Дата	Оценка
Студент	Шамаев И.Р.			
Принял	Гайнетдинова А.А			

Уфа 2019

Цель: Ознакомиться со сторонними библиотеками. Научиться работать с готовыми классами в C++

Теоретический материал

`window.clear();` – очищает экран.

`window.draw(shape);` – рисует объект.

`window.display();` – всё это показывает.

Из этого вытекает обязательный для SFML цикл:

```
while (window.isOpen())
{
    sf::Event event;
    while (window.pollEvent(event))
    {
        if (event.type == sf::Event::Closed)
            window.close();
    }

    window.clear();
    window.draw(shape);
    window.display();
}
```

Несколько **способов загрузки текстуры**:

- 1) Загрузка файла текстуры из памяти с помощью метода `loadFromMemory();`
- 2) Загрузка из потока с помощью метода `loadFromStream();`
- 3) Загрузка из файла, который уже был загружен, с помощью метода `loadFromImage();`

Индивидуальное задание №1

Задание: Написать приложение с использованием библиотеки SFML: В графическом окне случайным образом размещается от 3 до 10 смайликов с тремя различными эмоциями: улыбка (дуга, выгнутая вниз), недовольство (дуга выгнутая вверх), нейтральное (прямая линия). Смайлики перемещаются по окну, отталкиваются друг от друга и от стенок окна. При столкновении смайлики меняются эмоциями.

Описание программы:

Вводим количество смайлов и проверяем, чтобы их было больше 3 и меньше 10. Вводим текстуры на объекты. В цикле, задаем условия, при котором смайлы отталкиваются от стен и меняются текстурами при столкновении.

Исходный код программы

```
#include <iostream>
#include <SFML\Graphics.hpp>
#include <time.h>
#define Box_X 1000
#define Box_Y 900
// #define _CRT_SECURE_NO_WARNINGS

using namespace std;
using namespace sf;

int main()
{
    setlocale(LC_ALL, "Russian");
    srand(time(NULL));
    int n; // Количество смайлов

    cout << " Пожалуйста введите n = ";
    cin >> n;
    if (n < 3 || n > 10) // Проверка на корректность ввода
    {
        cout << "\nОшибка! Нужно ввести от 3 до 10 смайлов\n";
        return 1;
    }

    Image smileImage[11];
    Texture smileT[11];
    Sprite smileS[11];
    float vx[11], vy[11];
    int tex[11];

    for (int i = 0; i < n; i++)
    {
        vx[i] = ((double)(rand()) / RAND_MAX) + 1;
        vy[i] = ((double)(rand()) / RAND_MAX) + 1;
        float x = rand() % 800; // случайное появление
        float y = rand() % 800; // случайное появление
        if (i % 3 == 0) { smileImage[i].loadFromFile("U.png"); }
        else
        {
            if (i % 3 == 1) { smileImage[i].loadFromFile("P.png"); }
        }
    }
}
```

```

        else { smileImage[i].loadFromFile("N.png"); }
    }
    smileT[i].loadFromImage(smileImage[i]);
    smileS[i].setTexture(smileT[i]);
    smileS[i].setPosition(x, y);

    tex[i] = i % 3;
}

RenderWindow window(VideoMode(Box_X, Box_Y), "Smale");
window.setFramerateLimit(120);
while (window.isOpen())
{
    Event event;
    while (window.pollEvent(event))
    {
        if (event.type == Event::Closed)
            window.close();
    }
    window.clear(Color::White);
    for (int i = 0; i < n; i++)
    {
        window.draw(smileS[i]);

        if ((smileS[i].getPosition().x + vx[i] > Box_X - 60) ||
(smileS[i].getPosition().x + vx[i] < 0)) // Отскок смайла от рамки по x
        {
            vx[i] *= -1;
        }
        if ((smileS[i].getPosition().y + vy[i] > Box_Y - 60) ||
(smileS[i].getPosition().y + vy[i] < 0)) // Отскок смайла от рамки по y
        {
            vy[i] *= -1;
        }
        smileS[i].move(vx[i], vy[i]);
        for (int j = i + 1; j < n; j++) // смена смайлов
        {
            Vector2f v(smileS[i].getPosition() -
smileS[j].getPosition());
            if (v.x * v.x + v.y * v.y < 4 * 900)
            {
                int t = tex[i];
                tex[i] = tex[j];
                tex[j] = t;

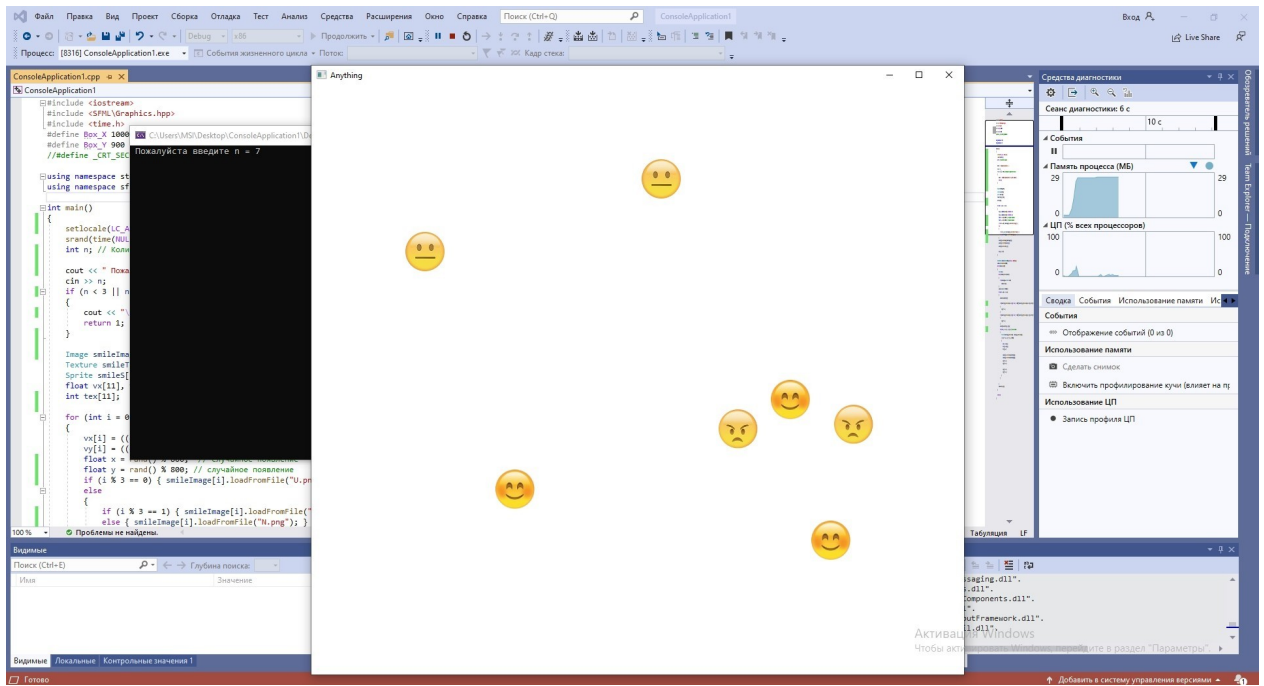
                smileS[i].setTexture(smileT[tex[i]]);
                smileS[j].setTexture(smileT[tex[j]]);
                vx[i] *= -1;
                vy[i] *= -1;

                vx[j] *= -1;
                vy[j] *= -1;
            }
        }
    }
    window.display();
}

return 0;
}

```

Пример выполнения программы



Вывод

Я научился работать с сторонними библиотеками и готовыми классами в C++