Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Курсовой проект по дисциплине:

«Программирование»

Пояснительная записка к курсовой работе

Тема работы:

«2D-платформер Horror game»

Исполнитель

студент гр. 653502 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Макаренко А.Д.

(подпись дата)

Руководитель

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Козуб В.Н.

(подпись дата)

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

(оценка)

Минск

2017 год

**Содержание**

[Введение 3](#_Toc480311999)

1. Общие сведения……………………………………………………………..4

2[. Сведения о программе](#_Toc480312000) 5

2[.1 Описание](#_Toc480312001) 5

2[.2 Содержание](#_Toc480312002) 5

3[. C++ и ООП](#_Toc480312004) 9

4[. Библиотека SFML](#_Toc480312008) 10

4[.1 Описание](#_Toc480312009) 10

4[.2 SFML на других языках программирования 10](#_Toc480312014)

4.3 [Модули](#_Toc480312014)11

Заключение12

[Список использованных источников](#_Toc480312026) 13

## Введение

Толчком к написанию данной курсовой работы послужило мое увлечение игровой индустрии. Поэтому я решил написать игру, т. к. это мой первый опыт, то выбрал 2D-платформер. С самого детства я играл в игры и любил всякие ужасы и прочие жуткие вещи. Из-за этого жанр игры — horror.

Целью данной курсовой работы является изучение строения кода 2D-платформеров и изучение библиотеки SFML.

Я выбрал SFML т. к. это свободная кроссплатформенная мультимедийная библиотека, простая в своем изучении.

В качестве языка программирования используется С++.

Целями и задачами курсового проектирования являются:

\* закрепление и нахождение новых знаний среды разработки С++;

\* использование дополнительной литературы;

\* изучение возможностей, предоставляемых библиотекой SFML;

## 1.Общие сведения

Программа была разработана учащимся факультета «ФКСиС» специальности «ИиТП» группы 653502 УО «Белорусский Государственный Университет Информатики и Радиоэлетроники» Макаренко Андреем Дмитриевичем в рамках курсового проекта в 2017 году.

Цель реализации задачи- создание игровой программы, а также знакомство с основными принципами работы с SFML.

## 2. Сведения о программе

## 2.1 Описание

Название игры - «Подвальный Фрэнк».

Действия игры происходят на дне загадочного мира под названием «Подвал». Подвал — это очень темное и страшное место, где происходят разного рода события, да еще и повторяющиеся постоянно. Ужас какой-то.

Цель: Выяснить, что же там забыл Фрэнк, прогуливаясь по замечательным уголкам этого замечательного подвала. А потом еще и попытаться найти выход и выбраться оттуда. А чтобы выбраться, вам нужно бегать и прыгать, преодолевая препятствия и прочее. И конечно же насладиться прекрасной графикой и пейзжами игрового мира, качеством прорисовки текстур и самого персонажа.

Для контроля персонажа используются стрелки.

## 2.2 Содержание

## Единственное, а может быть и нет, живое существо в нашем мире это Фрэнк. Соответственно единственный класс который присутствует «Player». То есть класс персонажа.

class PLAYER

{

public:

float dx, dy;

FloatRect rect;

bool onGround;

Sprite sprite;

}

dx, dy – переменные расположения персонажа, то есть координаты.

Rect – это фигурка персонажа.

OnGround – находиться ли персонаж на земле.

Sprite – это спрайт, переменная в которой хранятся текстуры персонажа.

Класс «Player» содержит определенные функции, такие как передвижение персонажа, прыжок и взаимодействие с миром, в котором он существует.

**Функция передвижения и прыжка, отвечающая за месторасположение персонажа:**

void update(float time)

{

rect.left += dx\*time;

Collision(0);

if (!onGround) dy = dy + 0.0005\*time;

rect.top += dy\*time;

onGround = false;

Collision(1);

currentFrame += 0.005\*time;

if (currentFrame > 12) currentFrame -= 12;

if(dx>0) sprite.setTextureRect(IntRect(32 \* int(currentFrame), 64, 32, 32));

if(dx<0) sprite.setTextureRect(IntRect(32 \* int(currentFrame) + 32, 64, -32, 32));

sprite.setPosition(rect.left-offsetX,rect.top-offsetY);

dx = 0;

}

**Функция взаимодействия персонажа с миром:**

void Collision(int dir)

{

for (int i = rect.top / 32; i<(rect.top + rect.height) / 32; i++)

for (int j = rect.left / 32 ;j<(rect.left + rect.width) / 32; j++)

{

if (TileMap[i][j] == 'B')

{

if ((dx>0) && (dir == 0)) rect.left = j \* 32 - rect.width;

if ((dx<0) && (dir == 0)) rect.left = j \* 32 + 32;

if ((dy>0) && (dir == 1)) { rect.top = i \* 32 - rect.height; dy = 0; onGround = true; }

if ((dy<0) && (dir == 1)) { rect.top = i \* 32 + 32; dy = 0; }

}

if (TileMap[i][j] == '0')

{

TileMap[i][j] = ' ';

monets++;

}

if (TileMap[i][j] == 'V')

{

music.pause();

sc[k].play();

Scream();

music.play();

TileMap[i][j] = ' ';

}

if (TileMap[i][j] == 'C')

{

music.pause();

sc[k].play();

Scream();

music.play();

TileMap[i][j] = ' ';

}

if ((TileMap[i][j] == 'E') && (monets > 0))

{

window.close();

RenderWindow end(VideoMode::getDesktopMode(), "End");

Texture tlast;

Sprite slast;

tlast.loadFromFile("end.png");

slast.setTexture(tlast);

while (end.isOpen())

{

Event event;

while (end.pollEvent(event))

{

if (event.type == Event::Closed)

end.close();

}

end.draw(slast);

end.display();

}

}

}

}

**Прорисовка карты** довольно таки проста, она состоит из символов, которым будут принадлежать определенные текстуры.

То есть со стороны кода она будет выглядеть так:

String TileMap[H] = {

"BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB",

"B 00 B",

"B B B",

"B BBBBBBBBBBBСBBBBBBCBBBBBBBBBBBBBBBBBBBBBB B",

"BBBB BB BBBB 00BB B B",

"B B C BB BBBB 00BB 0000 BBBB B",

"B BB VV B 000000 V 00 B",

"BC BBBBBBBBBBBBBBBBBBBBBBBB BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBVVB0B",

"B B 00 B",

"B BB BBBCBB00B BB V0 B",

"B B BB BCBB00B BB 000 B",

"BB B 0 BBC BCBB B BB B",

"B B 0 BBC VBB BB B BBBBBB B B",

"B BB B 0 BB BBBBB BBBBB B B 0000 BBBB B",

"BC BB V BB B CBB 0000 C B",

"BBBBBBBBBBBBBB BBBBBBBBBBBBBBBBBBBBBBBCCBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB",

"B B B B",

"BB BBBBBBBBBBBB BBB0 B",

"B B 00 BBBBB0 B",

"B B BBBBBBB0 B B",

"B B BB BBBBBBBBB0 B B VV B B",

"B B BBB 000 B B B0 B B B B0B BBB",

"B BB V BBBB B B B B B BV C BB0BB EB",

"BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB",

};

**И ее заполнение:**

if (TileMap[i][j] == 'B') { rectangle.setTextureRect(IntRect(512, 64, 32, 32)); }

if (TileMap[i][j] == '0') { rectangle.setTextureRect(IntRect(160, 256, 32, 32)); }

if (TileMap[i][j] == ' ') { rectangle.setTextureRect(IntRect(640, 256, 32, 32)); }

if (TileMap[i][j] == 'C') { rectangle.setTextureRect(IntRect(512, 64, 32, 32)); }

if (TileMap[i][j] == 'V') { rectangle.setTextureRect(IntRect(640, 256, 32, 32)); }

if (TileMap[i][j] == 'E') { rectangle.setTextureRect(IntRect(704, 192, 32, 32)); }

**Взаимодействие кода с клавишами:**

if (Keyboard::isKeyPressed(Keyboard::Left))

{

p.dx = -0.2;

}

if (Keyboard::isKeyPressed(Keyboard::Right))

{

p.dx = 0.2;

}

if (Keyboard::isKeyPressed(Keyboard::Up))

{

if (p.onGround) { p.dy = -0.3; p.onGround = false;}

}

## 3. C++ и ООП

## **C++** — компилируемый, статически типизированный язык программирования общего назначения.

## Поддерживает такие парадигмы программирования, как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование. Язык имеет богатую стандартную библиотеку, которая включает в себя распространённые контейнеры и алгоритмы, ввод-вывод, регулярные выражения, поддержку многопоточности и другие возможности. C++ сочетает свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником — языком C, — наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования.

## C++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также развлекательных приложений (игр). Существует множество реализаций языка C++, как бесплатных, так и коммерческих и для различных платформ. Например, на платформе x86 это GCC, Visual C++, Intel C++ Compiler, Embarcadero (Borland) C++ Builder и другие. C++ оказал огромное влияние на другие языки программирования, в первую очередь на Java и C#.

## Синтаксис C++ унаследован от языка C. Одним из принципов разработки было сохранение совместимости с C. Тем не менее, C++ не является в строгом смысле надмножеством C; множество программ, которые могут одинаково успешно транслироваться как компиляторами C, так и компиляторами C++, довольно велико, но не включает все возможные программы на C.

## **Объе́ктно-ориенти́рованное программи́рование (ООП)** — методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования.

## Необходимо обратить внимание на следующие важные части этого определения: 1) объектно-ориентированное программирование использует в качестве основных логических конструктивных элементов объекты, а не алгоритмы; 2) каждый объект является экземпляром определенного класса; 3) классы образуют иерархии. Программа считается объектно-ориентированной, только если выполнены все три указанных требования. В частности, программирование, не использующее наследование, называется не объектно-ориентированным, а программированием с помощью абстрактных типов данных.

## 4. Библиотека SFML

## 4.1 Описание

SFML ([англ](https://ru.wikipedia.org/wiki/Английский_язык). Simple and Fast Multimedia Library — простая и быстрая мультимедийная библиотека) — свободная [кроссплатформенная](https://ru.wikipedia.org/wiki/Кроссплатформенное_программное_обеспечение) мультимедийная [библиотека](https://ru.wikipedia.org/wiki/Библиотека_(программирование)). Написана на [C++](https://ru.wikipedia.org/wiki/C%2B%2B), но доступна также для [C](https://ru.wikipedia.org/wiki/C_(язык_программирования)), [D](https://ru.wikipedia.org/wiki/D_(язык_программирования)), [Java](https://ru.wikipedia.org/wiki/Java), [Python](https://ru.wikipedia.org/wiki/Python), [Ruby](https://ru.wikipedia.org/wiki/Ruby), [OCaml](https://ru.wikipedia.org/wiki/OCaml), [.Net](https://ru.wikipedia.org/wiki/.NET_Framework) и [Go](https://ru.wikipedia.org/wiki/Go). Представляет собой объектно-ориентированный аналог [SDL](https://ru.wikipedia.org/wiki/Simple_DirectMedia_Layer).

SFML содержит ряд модулей для простого программирования игр и мультимедиа приложений. Исходный код библиотеки предоставляется под лицензией [zlib/png license](https://ru.wikipedia.org/wiki/Zlib_License).

## 4.2 SFML на других языках программирования



### 4.3 Модули

В настоящее время доступны следующие модули:

* System — управление временем и потоками, он является обязательным, так как все модули зависят от него.
* Window — управление окнами и взаимодействием с пользователем.
* Graphics — делает простым отображение графических примитивов и изображений, для своей работы требует модуль Window.
* Audio — предоставляет интерфейс для управления звуком.

Network — для сетевых приложений.

## 

## Заключение

В данной курсовой работе была разработана игра. Меня не очень удовлетворила библиотека SFML. Допустим весомый минус ее в том, что она не может воспроизводить видео. Так же очень часто возникали ошибки, которые взаимодействовали с картой. Для начала, как по мне, она подходит довольно таки неплохо. Я слегка внедрился в саму систему создания игр. Я не могу сказать, что бесполезно потратил свое время. В процессе создания и я приобрел знания, которые хотел приобрести. Даже появилось желание сделать что-нибудь более стоящее чем платформер.

## Список использованных источников

1. https://www.sfml-dev.org/
2. <https://tlgrm.ru/docs/bots>
3. <https://ru.wikipedia.org/wiki/Ruby>
4. <https://www.ruby-lang.org/ru/libraries/>
5. <https://core.telegram.org/bots/api>
6. https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D0%B8%D0%B5%D0%BD%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5\_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5
7. <https://github.com/atipugin/telegram-bot-ruby>
8. <https://ru.wikipedia.org/wiki/Объектно-ориентированное_программирование>
9. <https://ru.wikipedia.org/wiki/Redis>
10. <https://github.com/redis/redis-rb>
11. <https://ru.wikipedia.org/wiki/VPS>
12. https://ru.wikipedia.org/wiki/C%2B%2B
13. [https://ru.wikipedia.org/wiki/SFML#SFML\_.D0.BD.D0.B0\_.D0.B4.D1.80.D1.83.D0.B3.D0.B8.D1.85\_.D1.8F.D0.B7.D1.8B.D0.BA.D0.B0.D1.85\_.D0.BF.D1.80.D0.BE.D0.B3.D1.80.D0.B0.D0.BC.D0.BC.D0.B8.D1.80.D0.BE.D0.B2.D0.B0.D0.BD.D0.B8.D1.8F](https://ru.wikipedia.org/wiki/SFML" \l "SFML_.D0.BD.D0.B0_.D0.B4.D1.80.D1.83.D0.B3.D0.B8.D1.85_.D1.8F.D0.B7.D1.8B.D0.BA.D0.B0.D1.85_.D0.BF.D1.80.D0.BE.D0.B3.D1.80.D0.B0.D0.BC.D0.BC.D0.B8.D1.80.D0.BE.D0.B2.D0.B0.D0.BD.D0.B8.D1.8F)
14. https://www.youtube.com/