Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Петрозаводский государственный университет»

Физико-технический институт

Кафедра информационно-измерительных систем и физической электроники

Отчёт по курсовому проекту по дисциплине «Технология программирования»

Работу выполнил:

студент группы 21312

Д.В.Пушко

Проверил:

канд. физ.-мат. наук

А.В. Бульба

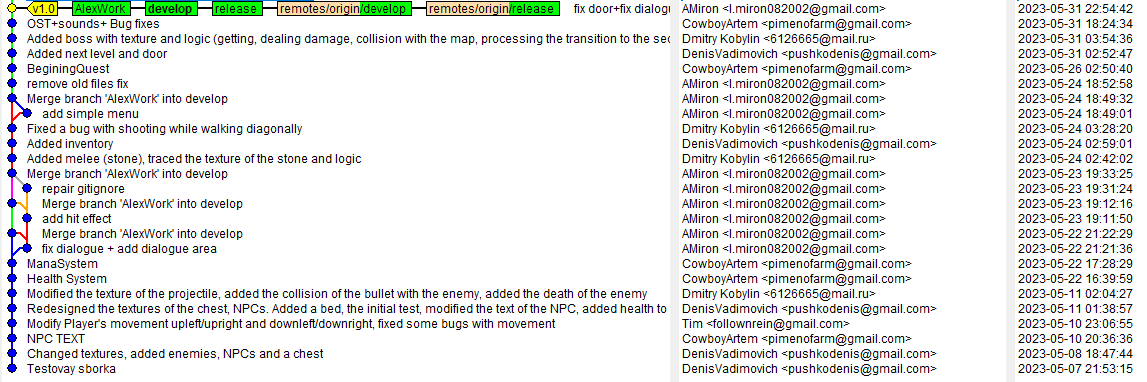
Петрозаводск 2023

**Цель работы.**

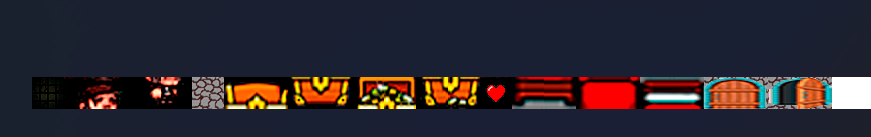
Используя возможности языка программирования С++ (IDE Qt Creator (Qt 5.4.2)) и библиотеки (Simple and Fast Multimedia Library) написать простую 2D игру.

В качестве примера был выбрана игра в стиле средневековой RPG, под названием «The Witcher 4».

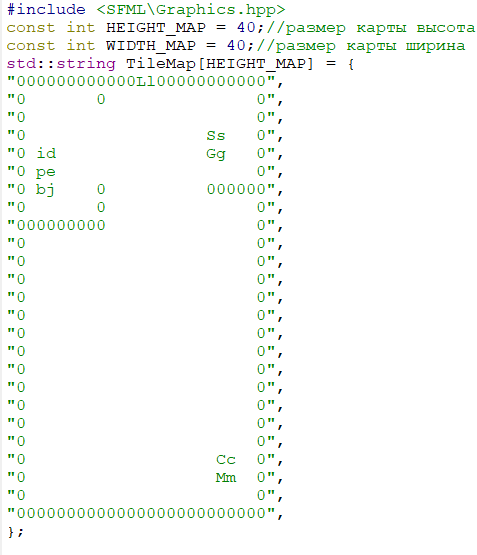
История коммитов:

****

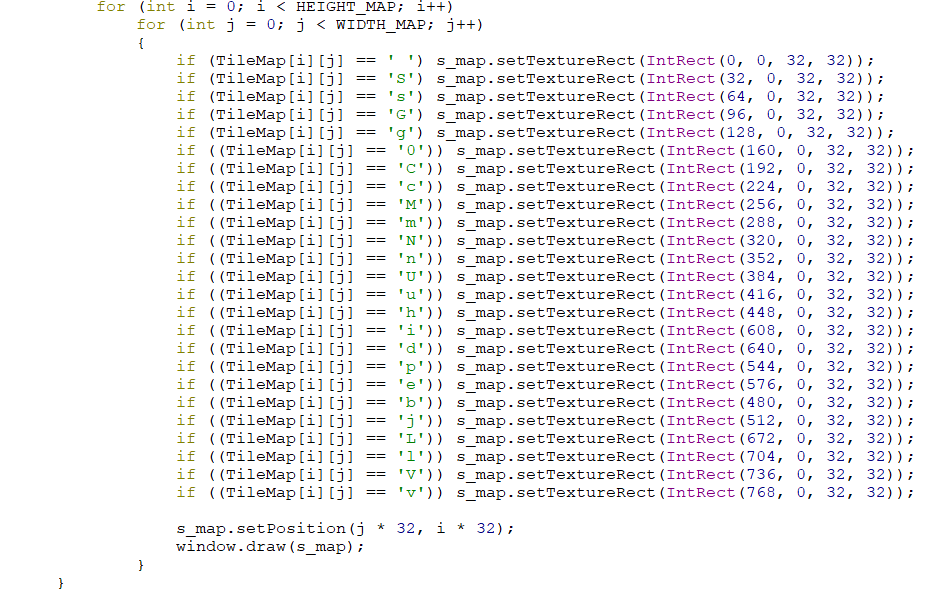
Вставил скелет игры, взятый с методического пособия и поменял текстуры, чтобы они были больше, был выбран метод разделение картинки на 4 фрагмента 32\*32.



Заголовочный файл map.h:



После в файле main.cpp, были взяты символы, отвечающие за каждый отдельный фрагмент и подставлены таким образом, чтобы текстура выглядела целостно.

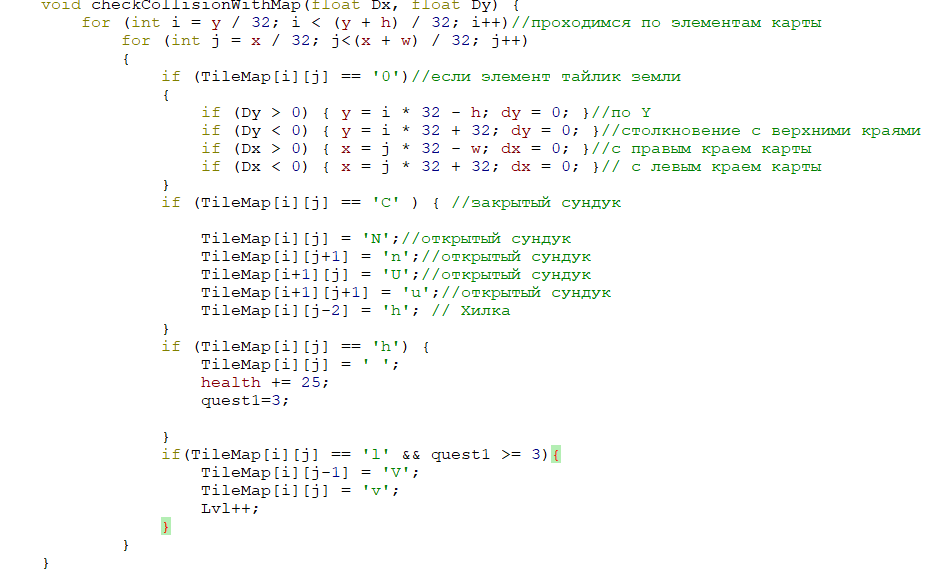


На скриншоте выше, представлен фрагмент кода файла main.cpp, который отвечает за загрузку нужного фрагмента текстуры в место, где поставлен символ.

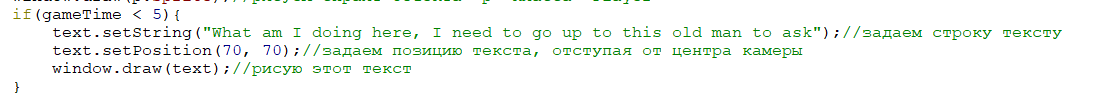


Так выглядит изображение игры, после загрузки текстур.

Была проработана коллизия игрока с текстурами.



На скриншоте выше, я добавил коллизию игрока с текстурой (символом, отвечающий за текстуру), при соприкосновении с символом «C» (отвечающий за левый верхний край сундука), текстура сундука меняется с закрытого состояния в открытое и на карту добавляется сердечко, при соприкосновении с которым, игроку добавляется 25 здоровья, а также переменная quest1, меняется на значение 3, это необходимо для дальнейшего прохождения квеста игры. После идет проверка на столкновение с текстурой двери, если он уже выполнил квест и подошел к двери, она меняется на состоянии открытой и к переменной «Lvl» прибавляется единица.



Так же была небольшая работа с текстом, которая заключается в том, что как только мы начинаем нашу игру, то некоторое время (5 секунд), у нас выводится строка, в которой наш игрок говорит «What am I doing here, I need to go up to this old man to ask», в этот момент и начинается игра.

Мной был проработан инвентарь, который заключался, в перемещении итератора по контейнеру Inventory.



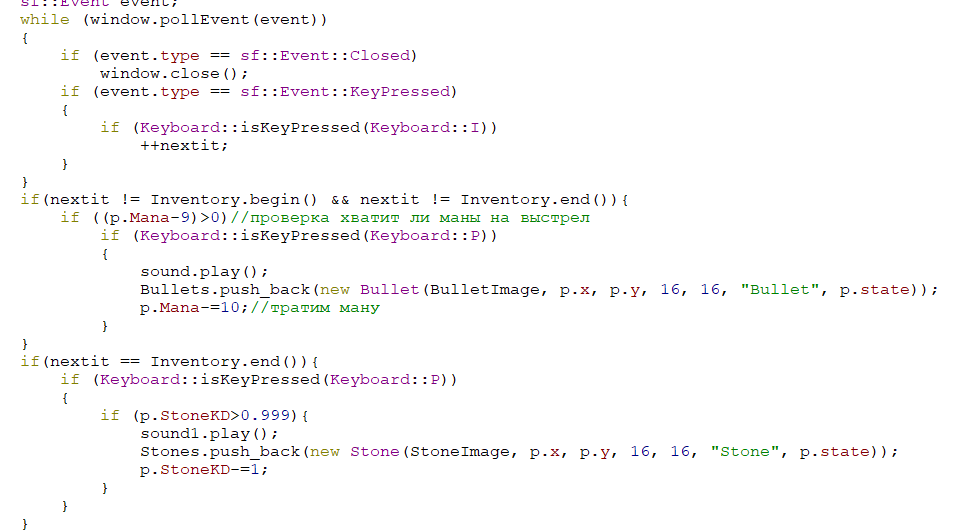
Для этого был создан список целочисленного типа, в котором было 3 значения, отвечающий за определенный предмет в руке игрока.



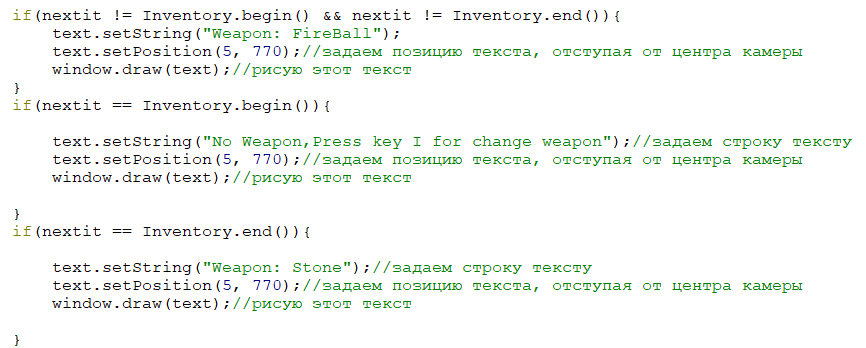
Создан итератор целочисленного типа, для работы с списком «инвентарь»,



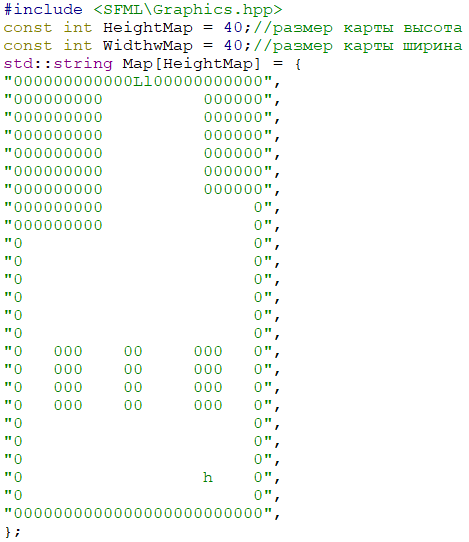
А также ему присвоено значения начало нашего списка.

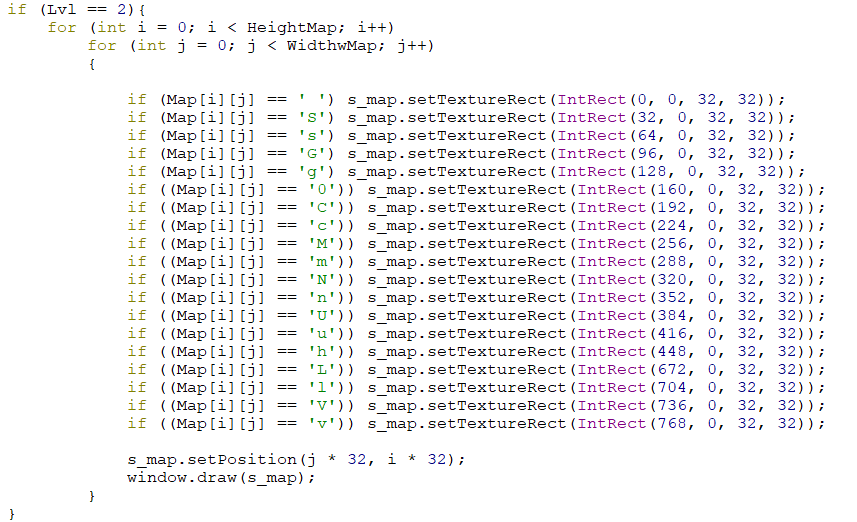


В скриншоте выше, представлен фрагмент кода, отвечающий за переключение предмета, а также за использование этого предмета. При нажатии кнопки «I», у нас итератор переключается на следующий элемент списка, и если итератор находится не в начала и не в конце списка, то у нас выбран предмет «FireBoll»(именуемый пулей), после нажатия кнопки «P», у нас вызывается список пули и добавляет элемент в конец списка инициализируя объект пуля, но это все происходит только если значение маны равно 10, соответственно после выстрела, значение маны становится равное нулю, но со временем восстанавливается. Такой же принцип работы и со вторым элементом списка «Stone», только если наш итератор равен концу списка. Если же наш итератор находится в начале списка, то выбранный элемент, характеризуется как «NoWeapon».

Для лучшего понимания, какой элемент выбран на данный момент, была проведена работа с выводом текста, которая по достижению того или иного элемента списка итератором, выводился необходимый текст



Финальной задачей моей работы над проектом, было добавления нового уровня в нашей игре. Реализовано с помощью глобальной переменной «Lvl», которой изначально присвоено значение 1 и в случае выполнения квеста и подходу к двери, менялось на значение 2, после чего подгружалось новое расположение фрагментов текстур, соответственно и вызывался массив с нового файла map1.h, который и содержал в себе расположение элементов. 



При достижении переменной «Lvl» состояния 2, загружаются фрагменты текстур, в место, где встречается символ.

Вывод: В ходе работы, мною были затронуты многие аспекты программирования игры, от работы с текстурами до работы с списком.