МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Балтийский государственный технический университет «ВОЕНМЕХ» им. Д.Ф. Устинова»
(БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова»)

БГТУ.СМК-Ф-4.2-К5-01

Факультет О «Естественнонаучный»

Кафедра О7 «Информационные системы и программная инженерия»

Дисциплина Программирование на языке высокого уровня

Курсовая работа
на тему:

Объектно-ориентированная разработка программ с графическим пользовательским интерфейсом «снизу-вверх».

Выполнил студент группы И407Б

Кузнецов Максим Павлович
Фамилия И.О.

ПРЕПОДАВАТЕЛЬ

Васюков В.М. _____
Фамилия И.О. Подпись

Оценка _____

«_____» _____ 2021 г.

Санкт-Петербург
2021 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Цель работы – создать игру «Снег на голову», ознакомиться с работой библиотеки SDL2 [1], понять основные принципы ООП на практике.

SDL – Simple DirectMedia Layer – это свободная кроссплатформенная мультимедийная библиотека, реализующая единый программный интерфейс к графической подсистеме, звуковым устройствам и средствам ввода для широкого спектра платформ. Данная библиотека активно используется при написании кроссплатформенных мультимедийных программ (в основном игр). Официально она поддерживает такие платформы, как: Windows, Linux, Mac OS, iOS и Android. Сам же SDL предоставляет огромное количество инструментов, для работы с видео, изображениями, аудио и взаимодействием с клавиатурой. Основные возможности SDL – поддержка операций над двумерными плоскостями пикселей (включая создание окон), обработка событий (от клавиатуры, мыши и таймера), а также работа со звуком и абстрагирование доступа к файлам. Непосредственно SDL содержит лишь весьма ограниченный, базовый набор возможностей, а дополнительные функции (такие, как графические примитивы или вывод текста) обеспечиваются расширениями библиотеки.

SDL сам по себе довольно прост. Его можно рассматривать как тонкую прослойку, обеспечивающую поддержку для 2D-операций над пикселями, звука, доступа к файлам, обработки событий и т. п. Он часто используется в дополнение к OpenGL, предоставляя поддержку мыши, клавиатуры и джойстиков.

Библиотека состоит из нескольких подсистем, таких как Video, Audio, CD-ROM, Joystick и Timer. В дополнение к этой базовой низкоуровневой

функциональности, существует ряд стандартных библиотек, предоставляющих дополнительную функциональность:

- SDL_image — поддержка различных растровых форматов

- SDL_mixer — функции для организации сложного аудио, в основном, сведение звука из нескольких источников

- SDL_net — поддержка сетевых функций

- SDL_ttf — поддержка шрифтов TrueType

- SDL_rtf — отрисовка текста в формате RTF (доступна только для SDL 1.2)

Игра – вариант № 5: С неба (верхняя часть экрана) сыплется очень крупный град. Управляя человечком(стрелками или клавишами “A” и “D”), игрок должен избежать попадания градин по нему. С течением времени плотность града увеличивается. Игра продолжается до тех пор, пока одна из градин не упадет на человечка, после чего на экран выводится время игры.

1 Постановка задачи

Целью курсовой работы является создание компьютерной игры на языке C++ [3] с использованием графического пользовательского интерфейса и библиотеки SDL 2.0 [1].

Название игры: «Снег на голову».

Правила игры: С неба (верхняя часть экрана) сыплется очень крупный град. Управляя человечком(стрелками или клавишами “A” и “D”), игрок должен избежать попадания градин по нему. С течением времени плотность града увеличивается. Игра продолжается до тех пор, пока одна из градин не упадет на человечка, после чего на экран выводится время игры.

Цель игры : «продержаться» как можно дольше времени.

2 Описание программы

2.1 Используемые файлы

Проект состоит из следующих файлов:

- snowfall.cpp – основной заголовочный файл;
- classes.h – заголовочный файл с классами;
- func.h – заголовочный файл с функциями;
- allura.ttf – файл шрифта;
- snow.png – изображение снежинки (градинки);
- bcg.png – изображение игрового фона;
- man_1.png – изображение игрока в первом положении;
- man_2.png – изображение игрока во втором положении;
- man_3.png – изображение игрока в третьем положении;
- man_4.png – изображение игрока в четвертом положении;
- man_5.png – изображение игрока в пятом положении;
- man_6.png – изображение игрока в шестом положении;

К программе подключаются библиотеки SDL2 [1], SDL2_image [2], SDL2_ttf [5].

2.2 Диаграмма классов

На рисунке 1 представлена схема связей между классами.

Обычная черная стрелка - наследование, пунктирная - зависимость.

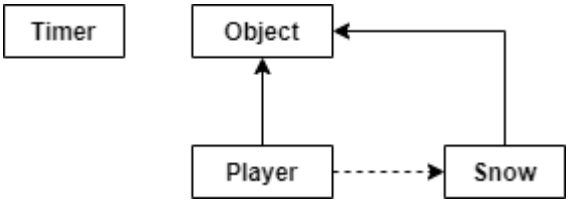

Рисунок 1 – Схема связей между классами

Далее представлены описания классов (их поля и методы).

На рисунке 2 представлено описание классов Object, Timer, Snow и

Player.

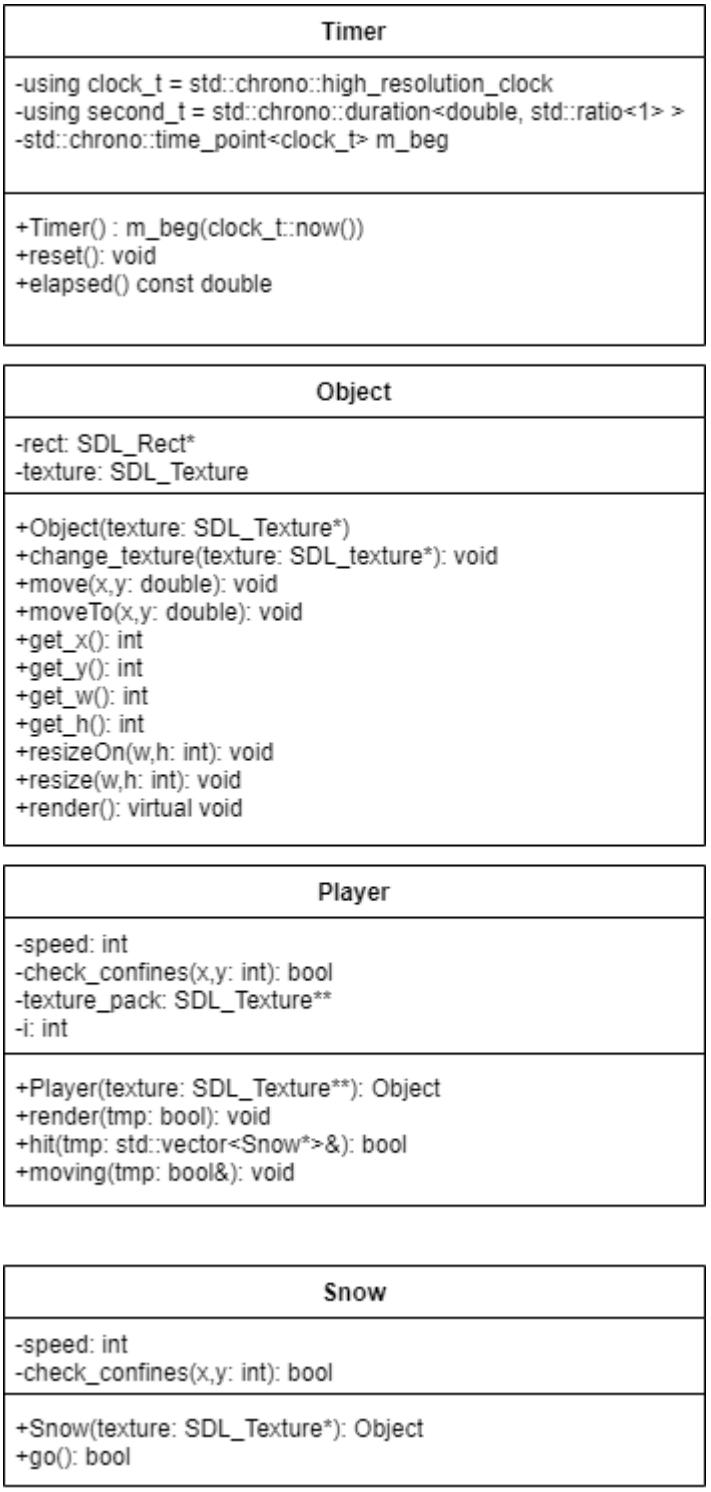

Рисунок 2 – Описание классов Object, Timer, Snow и Player

2.3 Структура меню

Структура меню представлена на рисунке 3.

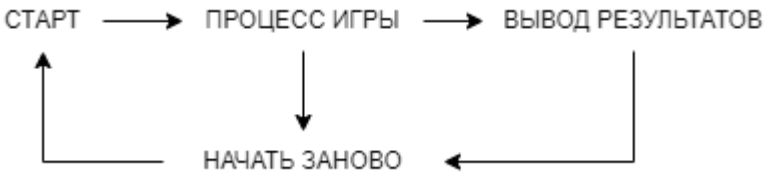

Рисунок 3 – Структура меню

2.4 Описание изображений

В программе используется шесть положений главного персонажа, которые сменяют друг друга при перемещении персонажа по игровому полю представленных на рисунке 4, рисунке 5, рисунке 6, рисунке 7, рисунке 8, рисунке 9.

Рисунок 4 – изображение первого положения персонажа

Рисунок 5 – изображение второго положения персонажа

Рисунок 6 – изображение третьего положения персонажа

Рисунок 7 – изображение четвертого положения персонажа

Рисунок 8 – изображение пятого положения персонажа

Рисунок 9 – изображение шестого положения персонажа

Также в программе используется изображение для фона, представленное на рисунке 10, и изображение снежинки, представленное на рисунке 11.

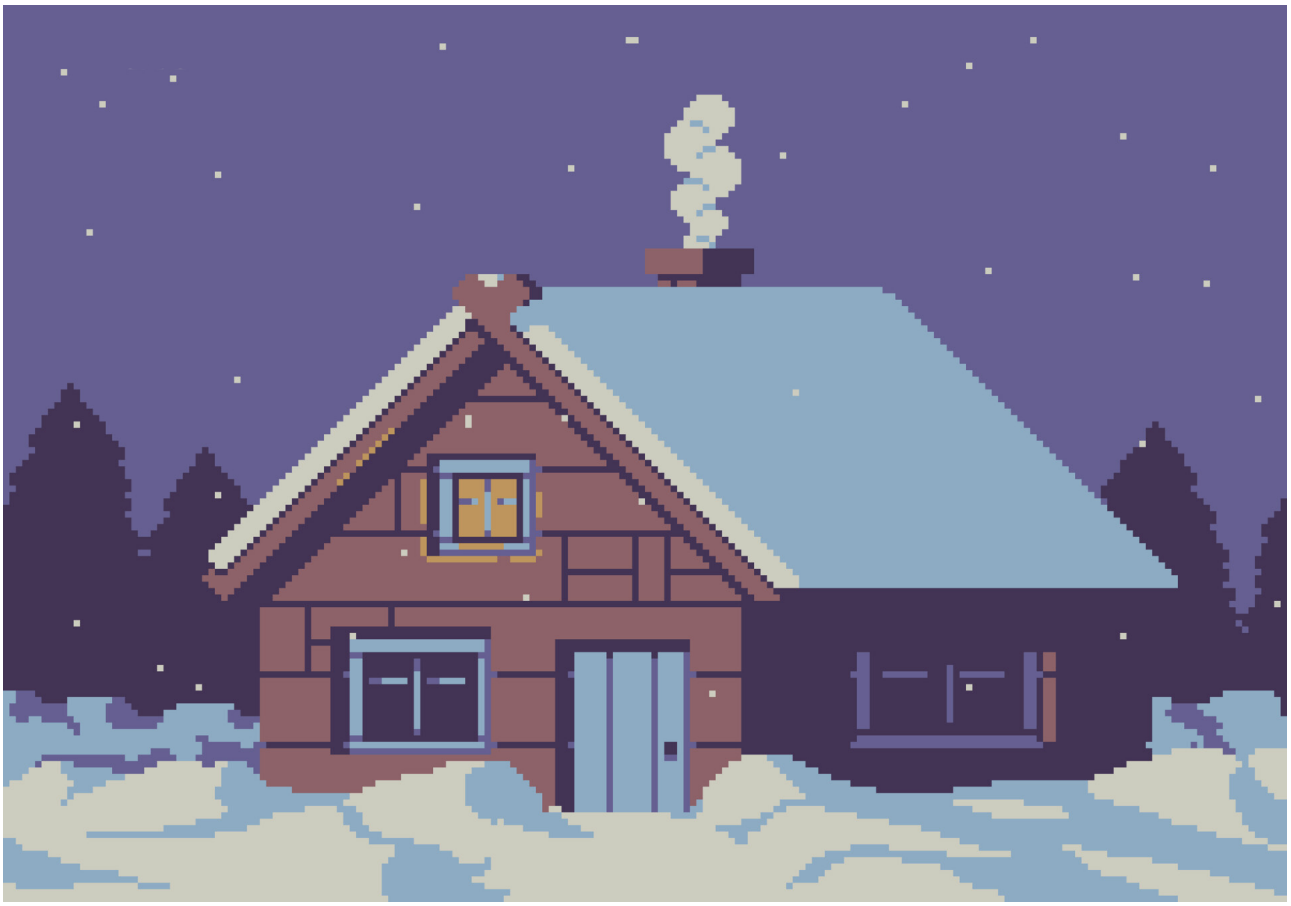

Рисунок 10 – Игровой фон

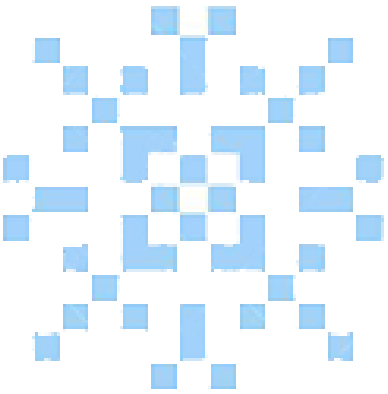

Рисунок 11 – Снежинка, от которой надо уворачиваться

2.5 Результаты работы программы

На рисунке 12 показан старт игры. Изначальное время на таймере равно 0.

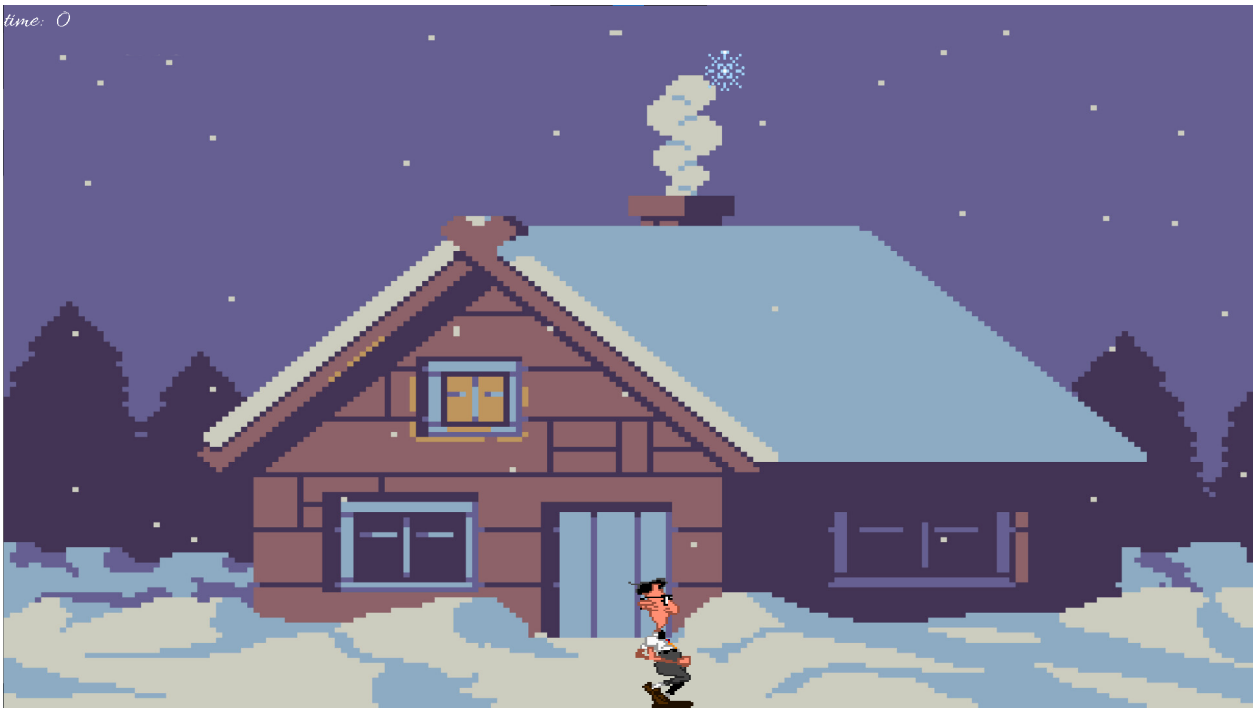

Рисунок 12 – старт игры

На рисунке 13 показано, что за время выживания прибавляется время, которое равно очкам игрока

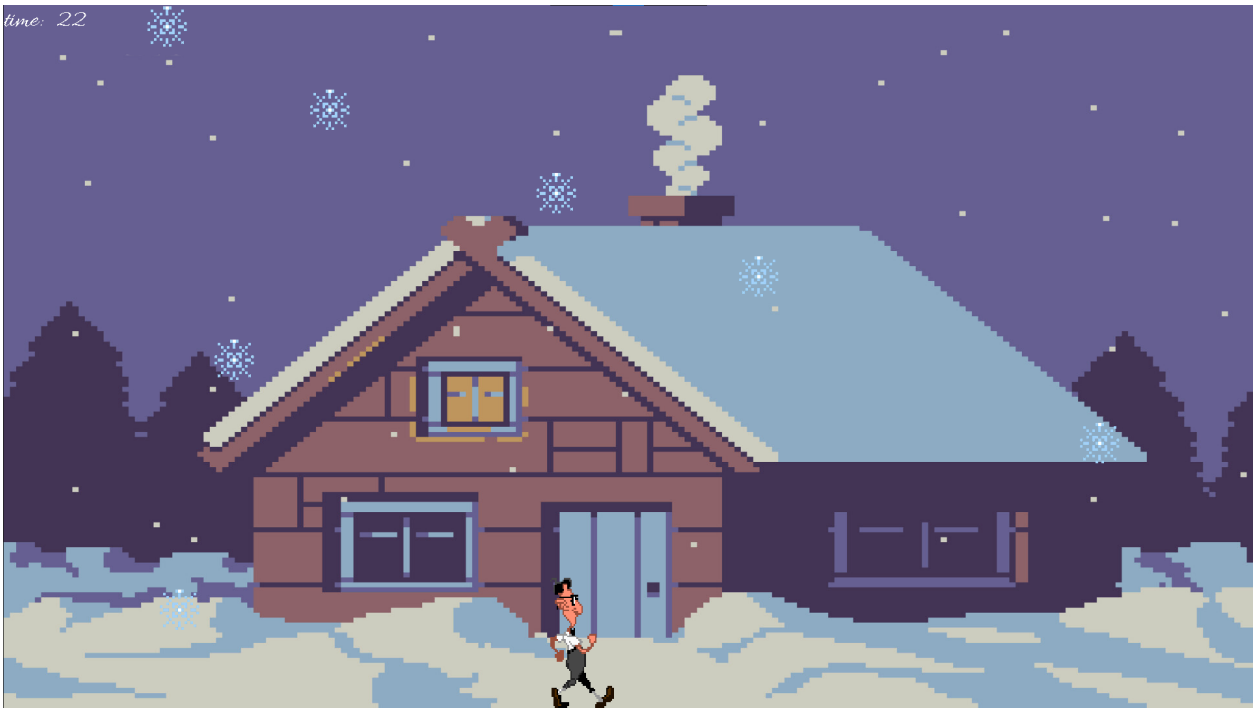

Рисунок 13 – Игровой процесс и начисление очков

На рисунке 14 показано, что с набором очков увеличивается и сложность игры, за счет увеличения числа снежинок, от которых игрок уворачивается

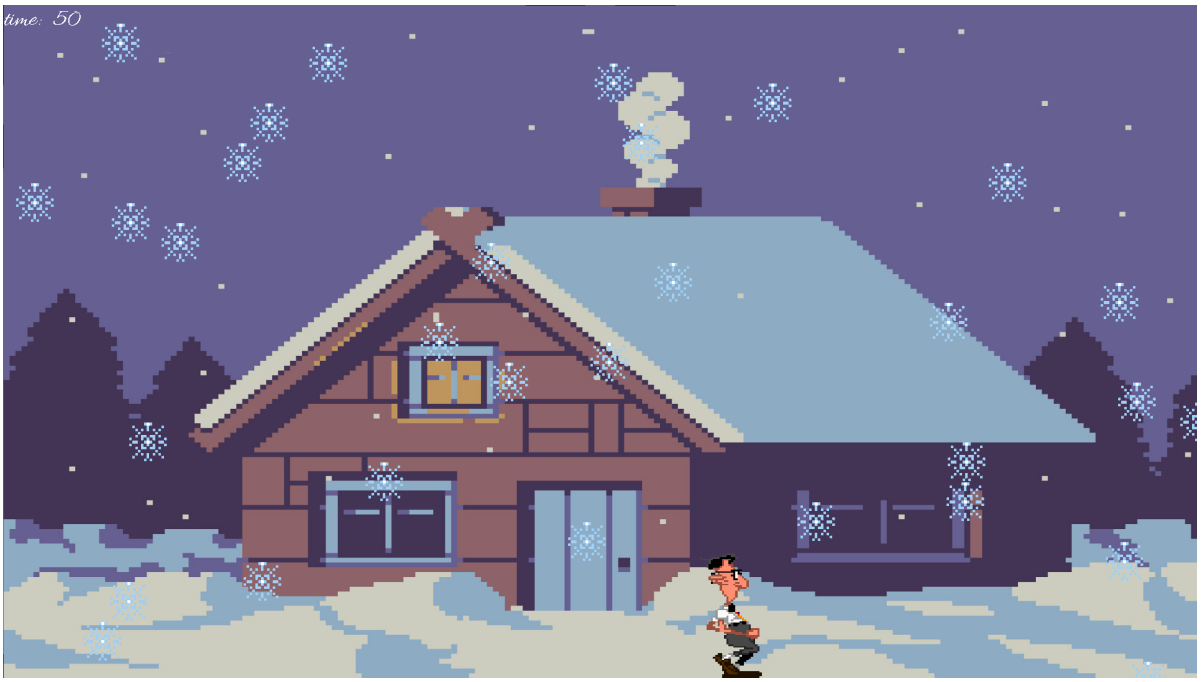

Рисунок 14 – Усложнение игрового процесса

На рисунке 15 показано, как выглядят результаты в случае поражения и таблицу лучших результатов за все игры.

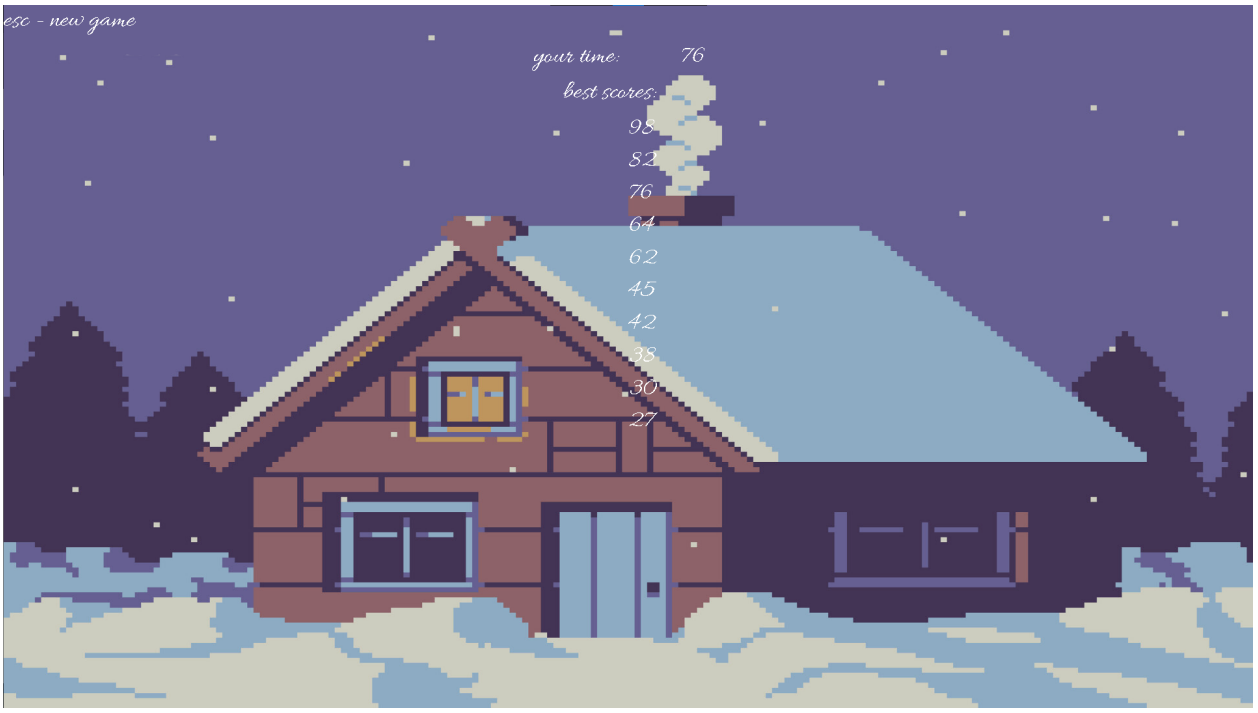

Рисунок 15 – Вывод результата игрока и лучших результатов

При нажатии клавиши “Esc” мы начинаем игру заново,очки сбрасываются и игровое поле очищается от снежинок, что отображено на рисунке 16.

Рисунок 16 – начало новой игры после показа лучших результатов

ЗАКЛЮЧЕНИЕ

В рамках курсовой работы была разработана игра на C++ [3] с использованием графической библиотеки SDL2 [1], SDL_Image [2] и SDL2_ttf [5]. В процессе выполнения курсовой работы были изучены динамические структуры данных, работа с бинарными файлами [4], использование классов и принципов ООП, в том числе инкапсуляции.

Была реализована сама игра, а также добавление информации о тараканах в бинарный файл и получение информации из него же.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Документация библиотеки SDL2. URL: https://wiki.libsdl.org/ (Дата обращения: 20.09.2021).

2. Документация библиотеки SDL_Image. URL: https://www.libsdl.org/projects/SDL_image/docs/index.html (Дата обращения: 20.09.2021).

3. Прата С. Язык программирования C++ - изд. «Вильямс», 2012 - 490 с.

4. Работа с бинарными файлами. URL: https://proginfo.ru/binary-file/ (Дата обращения: 20.09.2021).

5. Документация библиотеки SDL2_ttf. URL: https://www.libsdl.org/projects/SDL_ttf/docs/ (Дата обращения: 20.09.2021).

ПРИЛОЖЕНИЕ А

Исходный код программы и описание основного файла

Исходные файлы программы представлены на приложенном диске.