НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«Київський політехнічний інститут імені Ігоря Сікорського»

кафедра Технічної кібернетики

**Тема: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**UA.ІК.01310-01 01**

*Курсова робота*

дисципліна

*“Структури даних та алгоритми”*

Керівник: Виконав:

доц. Богданова Н.В. \_\_\_\_\_\_\_

“Захист дозволено” гр. ІК-9х

“\_\_\_” \_\_\_\_\_\_\_\_\_\_ 2021 р.

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

/підпис керівника/ /підпис студента/

Захищено з оцінкою

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

2021

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра Технічної кібернетики

Дисципліна Структури даних та алгоритми

Курс **I**  Група ІК-01 Семестр **II**

ЗАВДАННЯ

**на курсову роботу студента**

(прізвище, ім’я, по батькові)

1. Тема роботи Створення комп’ютерної гри «Battle City»

2. Строк здачі студентом закінченої роботи

3. Вихідні дані до роботи мова програмування

Інтерактивна віконна оболонка з реалізацією комп’ютерної гри «Battle City», що

повинна містити вікно привітання, головне меню, таблицю результатів, та власне

реалізовану комп’ютерну гру «Battle City».

Реалізація програмного коду до гри «Змійка» була здійснена за допомогою

об’єктно-орієнтованої мови програмування – C++

4. Зміст розрахунково – пояснювальної записки (перелік питань, які підлягають розробці)

Вступ; 1. Ігровий процес; 2. Постановка задачі; 3. Алгоритмічна Модель та

структура программи; 3.1 uml-схема класів; 4. Перелік і призначення режимів та

структура діалогу; 5. Структура даних та ресурсів програми; 5.1 класи програми;

6. Опис програми; 7. Інструкція адміністратора; 8. Керівництво користувача;

Висновки; використані джерела

5. Перелік графічного матеріалу ( з точним зазначенням обов’язкових креслень)

*Додатки (UML-діаграми, код програми, скріншоти).*

6. Дата видачі завдання

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |  |
| --- | --- | --- | --- |
| **№**  п\п | **Назва етапів курсової роботи** | **Строк виконання**  **етапів роботи** | **Примітки** |
| 1 | Отримання та узгодженя теми курсової роботи | 15.03.2021 |  |
| 2 | Визначення структури програмних модулів | 20.04.2021 |  |
| 3 | Визначення головних класів гри | 21.04.2021 |  |
| 4 | Розробка програмного коду | 23.04.2021 |  |
| 5 | Оптимізація програми | 7.05.2021 |  |
| 6 | Налагодження та тестування програми | 7.05.2021 |  |
| 7 | Оформлення роботи | 8.05.2021 |  |
| 8 | Здача на перевірку (електронна версія) | 10.05.2021 |  |
| 9 | Виправлення недоліків та доробка програми |  |  |
| 10 | Захист |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Студент

(прізвище, ім’я, по батькові студента)

Керівник Богданова Н.В.

( прізвище, ім’я, по батькові

викладача)

« » 2021 р.

ЗМІСТ

[ВСТУП 4](#_Toc71546717)

[1. ІГРОВИЙ ПРОЦЕС 5](#_Toc71546718)

[2. ПОСТАНОВКА ЗАДАЧІ 6](#_Toc71546719)

[3. АЛГОРИТМІЧНА МОДЕЛЬ ТА СТРУКТУРА ПРОГРАММИ 7](#_Toc71546720)

[3.1 UML-СХЕМА КЛАСІВ 8](#_Toc71546721)

[4. ПЕРЕЛІК І ПРИЗНАЧЕННЯ РЕЖИМІВ ТА СТРУКТУРА ДІАЛОГУ 9](#_Toc71546722)

[5. СТРУКТУРА ДАНИХ ТА РЕСУРСІВ ПРОГРАМИ 10](#_Toc71546723)

[5.1 КЛАСИ ПРОГРАМИ 10](#_Toc71546724)

[6. ОПИС ПРОГРАМИ 12](#_Toc71546725)

[7. ІНСТРУКЦІЯ АДМІНІСТРАТОРА 16](#_Toc71546726)

[8. КЕРІВНИЦТВО КОРИСТУВАЧА 17](#_Toc71546727)

[ВИСНОВКИ 18](#_Toc71546728)

[ВИКОРИСТАНІ ДЖЕРЕЛА 19](#_Toc71546729)

# ВСТУП

У ході роботи над проектом є:

1. Набути навиків використання основних принципів ООП при розробці програми мовою програмування C++
2. Набути навиків роботи з графічною бібліотекою SFML
3. Набути навиків роботи з бібліотекою парсингу данних JSON for Modern C++

# ІГРОВИЙ ПРОЦЕС

Гравець обирає кількість гравців (1-2) які будуть грати та початковий рівень гри. Танк контролюється за допомогою клавіш WASD та стріляє за допомогою пробілу (для другого гравця це клавіші стрілочки та правий ctrl). Якщо граветь знищив 20 ворожих танків то його переносить на наступний рівень. Мета набрати якнайбільше балів поки не закінчяться життя танка або не буде зруйнована його база. Після чого гра закінчується, підраховуються бали за цей рівень та виводиться таблиця рекордів. Якщо гравець набрав більше балів за 10 місце в таблиці йому пропонується ввести його ім’я.

# ПОСТАНОВКА ЗАДАЧІ

Цільовим напрямком курсової роботи є розробка комп’ютерної гри «Battle City». Гра має забезпечувати виконання таких умов і містити:

1. Головне меню з логотипом, з можливостями почати гру, переглянути
2. таблицю рекордів або вийти із гри.
3. Таблицю рекордів, де відображаються 10 найкращих результатів гравців.

Дивлячись на це, потрібно вирішити такі проблеми:

1. Намалювати потрібні для гри графічні об’єкти (спрайти);
2. Розробити алгоритм гри:
3. Рух танка по мапі з уникненням колізій як з самою мапою так і іншими танками, як фізичними об’єктоми;
4. Рух ворожих танків за алгоритмом схожим на алгоритм з оригінальної гри
5. Реалізувати зв’язок між класами;
6. Анімувати рухомі об’єкти на мапі;
7. Виконати зв’язок між сценами за допомогою графічного інтерфейсу;
8. Реалізувати збереження та завантаження таблиці рекордів

# 3. АЛГОРИТМІЧНА МОДЕЛЬ ТА СТРУКТУРА ПРОГРАММИ

Реалізація програмного коду для гри «Battle City» була здійснена за допомогою об’єктно-орієнтованої мови програмування C++ в середовищі розробки Microsoft Visual Studio.

Основним керуючим класом є Game, який ініціалізує всі ігрові об’єкти, виконує їх оновлення та вивід на екран.

Графічний інтерфейс був реалізований за допомогою графічної бібліотеки SFML, яка дає можливість керувати вікнами та виводи малювати на них. Кожен важливий клас має процедуру Draw, яка виводить об’єкти цього класу на екран.

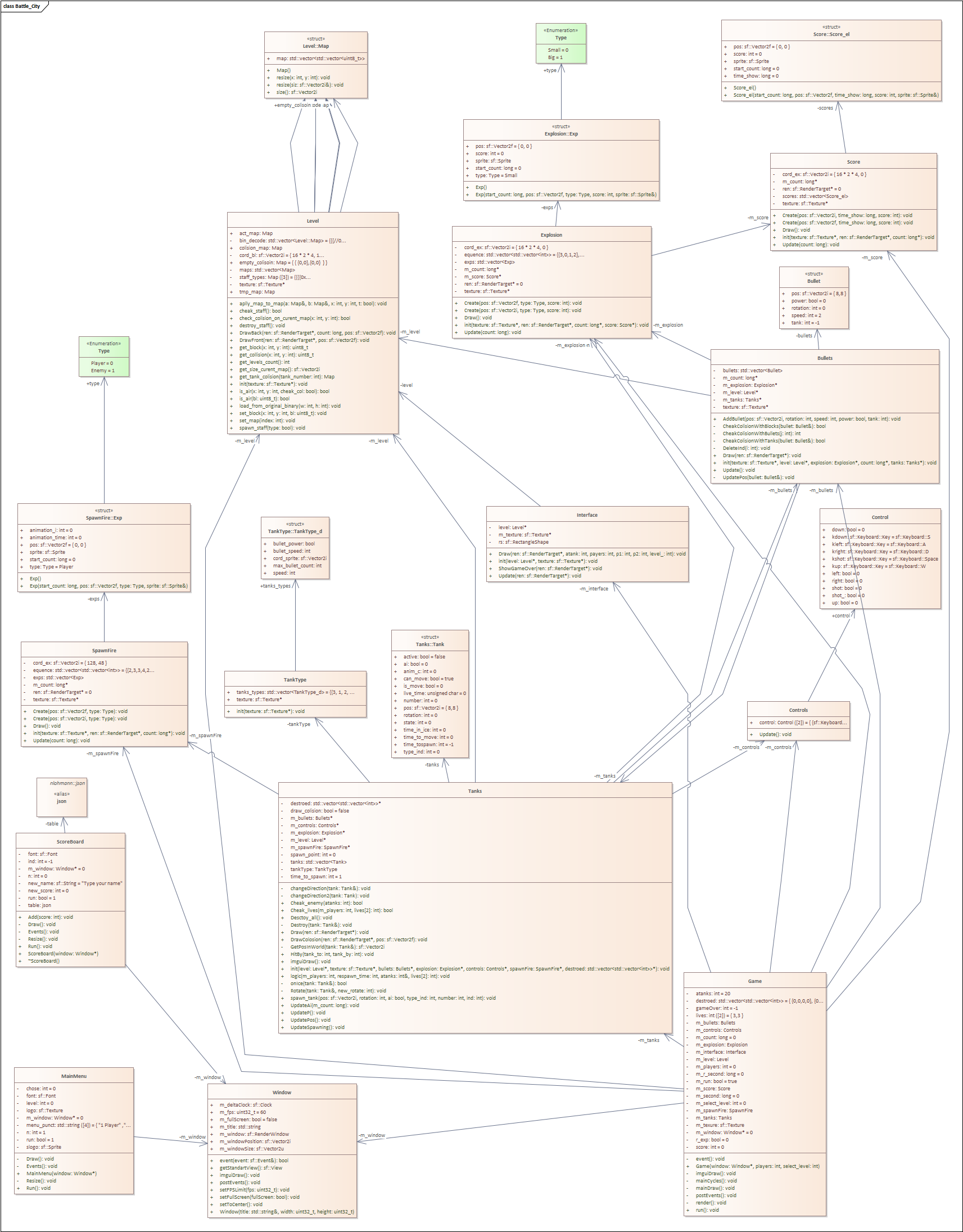
Пересування танків гравця відбувається за допомогою зміні координат в двовимірній системі. В класі Tanks в методі UpdatePos описана логіка пересування та перевірка на зіткнення з іншими об’єктами.

Пересування ворожих танків відбувається за допомогою алгоритму подібному до оригінального, він розбитий на 3 фази довжина яких залежить від рівня складності та кількості гравців. Фаза обирається за часом життя танка. На першій фазі рух танків випадковий, на другій вони починають переслідувати гравців, а на третій беруть напрям на базу гравця.

Мапа, як фізичний об’єкт являється квадратною матрицею, кожний елемент якої є типом блоку в цих координатах. Перевірка колізій між об’єктами виконується за допомогою перевірки типу блоку в координатах.

Таблиця рекордів реалізована за допомогою збереження файлу в форматі JSON.

## 3.1 UML-СХЕМА КЛАСІВ



# 4. ПЕРЕЛІК І ПРИЗНАЧЕННЯ РЕЖИМІВ ТА СТРУКТУРА ДІАЛОГУ

Дана програма має два режими роботи:

1. Без рівнів доступу користувача

2. В незалежності від рівня доступу користувача

При запуску програми перед користувачем з’являється віконна оболонка гри «Battle City» з логотипом та головним меню. Меню надає можливість обрати початковий рівень, розпочати гру, переглянути таблицю рекордів та вийти із гри. При натисненні кнопки 1 Player перед користувачем з’являється сама гра. Після завершення гри користувачу вісвітлюється таблиця рекордів, і якщо він набрав більше балів за останє місце, він може ввести своє ім’я. Далі він може повернутись до головного меню.

Діалог між системою та користувачем відбувається завдяки користувацькому, інтуїтивного інтерфейсу. Кожне меню має опис керування.

# 5. СТРУКТУРА ДАНИХ ТА РЕСУРСІВ ПРОГРАМИ

## 5.1 КЛАСИ ПРОГРАМИ

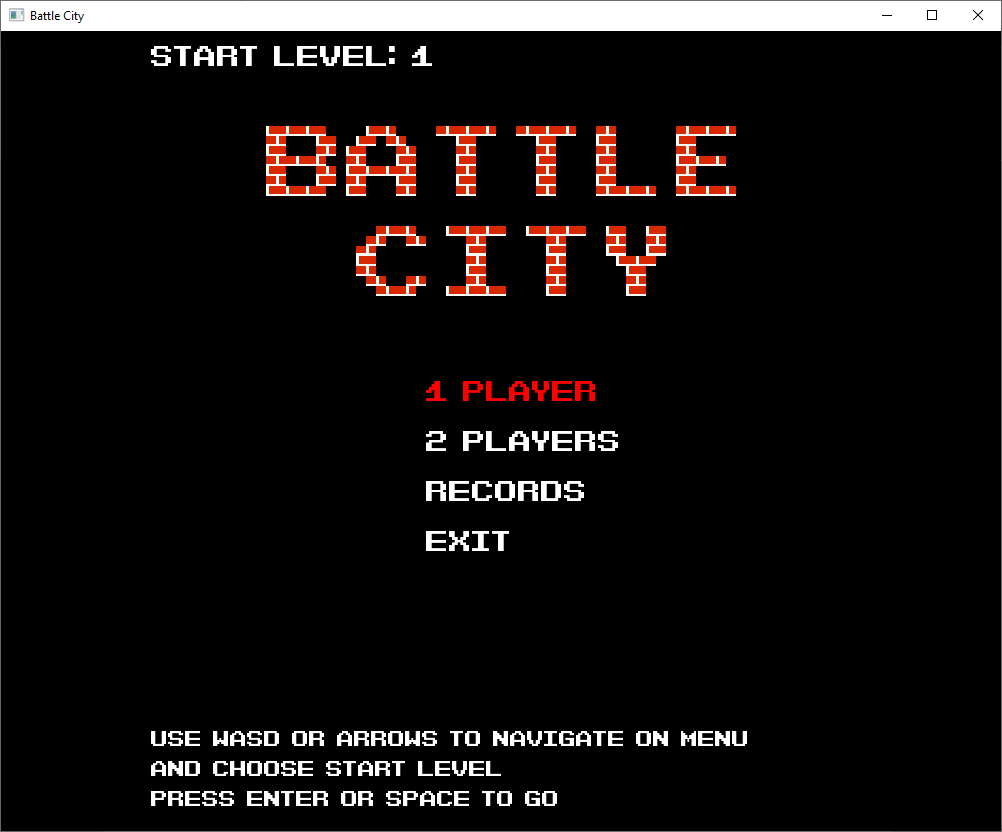
Для реалізації програмного коду використовуються такі класи, як: Bullets, Controls, Explosion, Game, Interface, Level, MainMenu, Score, ScoreBoard, SpawnFire, Tanks, TankType, Window.

|  |  |
| --- | --- |
| Клас | Використання |
| Bullets | Відповідає за обробку пуль: створення, оновлення, вивід на екран. Перевіряє та оброблює їх зіткнення з між собою, з танками, та з елементами миру. |
| Controls | Відповідає за керування танками за допомогою клавіатури |
| Explosion | Відповідає за створення та відображення вибухів |
| Game | Головний клас, що вміщує реалізацію гри і зв’язує всі головні класи. Також відповідальний за обробку виграшу та програшу |
| Interface | Відповідає за вивід інтерфейсу гри на екран |
| Level | Відповідальний за завантаження, обробку та зміну ігрового поля. Також оброблює події з головною базою гравця. |
| MainMenu | Відповідальний за керування головним меню та вибором початкового рівня |
| Score | Відповідальний за відображення зароблених балів за знищення ворожого танку |
| ScoreBoard | Відповідальний за вивід таблиці рекордів та додавання нового рекорду |
| SpawnFire | Відповідає за анімацію створення танків |
| Tanks | Відповідає за створення, пересування, зітхання з іншими танками та ігровим полем, видалення танків. Відповідає за штучний інтелект танків. |
| TankType | Зберігає інформацію про існуючі типи танків |
| Window | Відповідає за керування вікном гри |

# 6. ОПИС ПРОГРАМИ

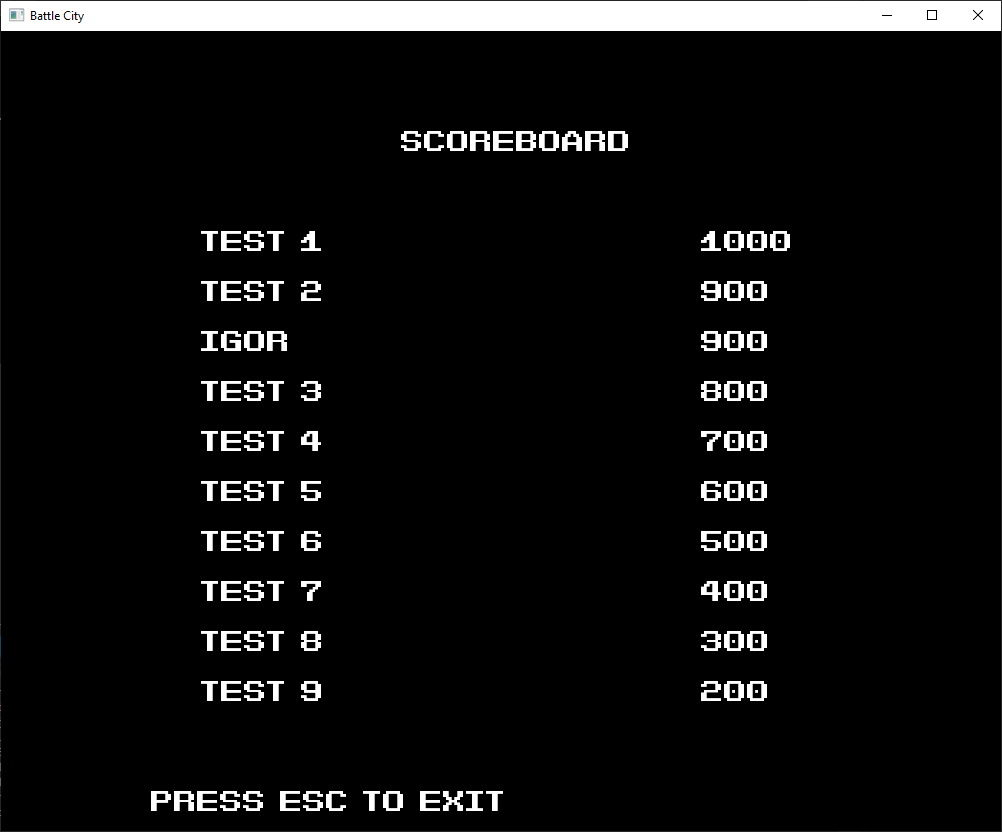
Для опису роботи програми будуть представлені фото роботи програми, де будуть описані основні елементи сцен та ігрового процесу.

Опис програми: Після запуску програми перед користувачем відкриється головне меню, яке складається з номеру початкового рівня, логотипу гри, опису керування та чотирьох кнопок: 1 PLAYER, 2 PLAYERS, RECORDS, EXIT. Меню керуються за допомогою клавіатури.

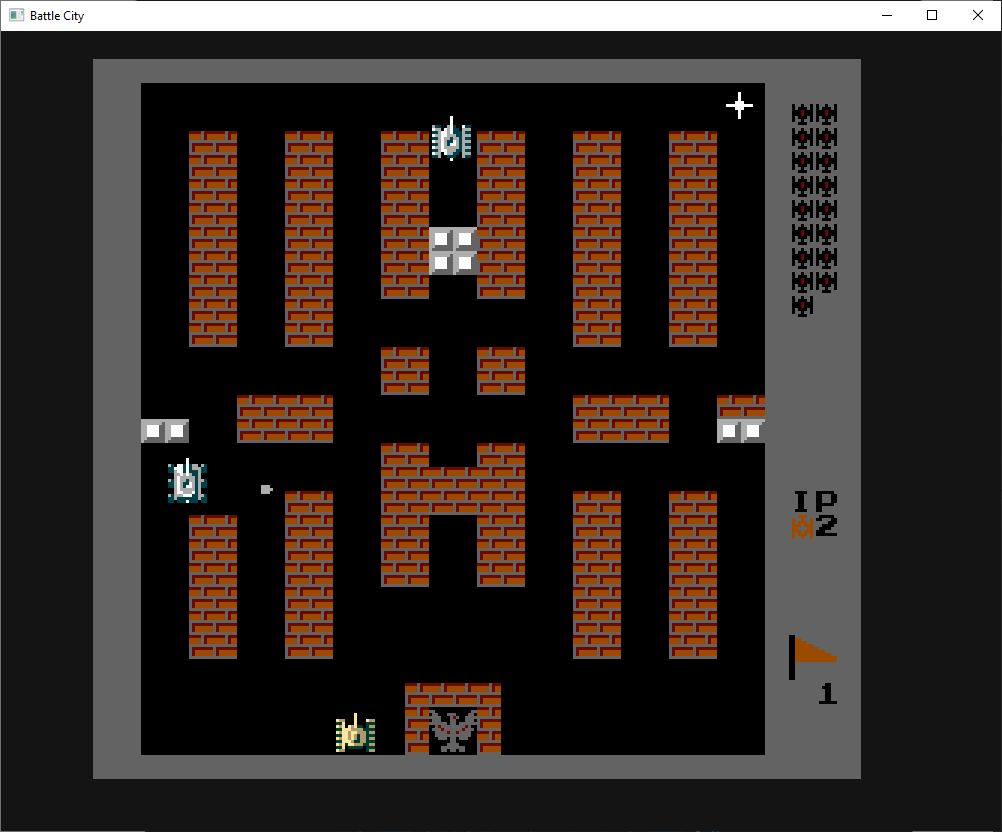


6.1 Головне меню гри

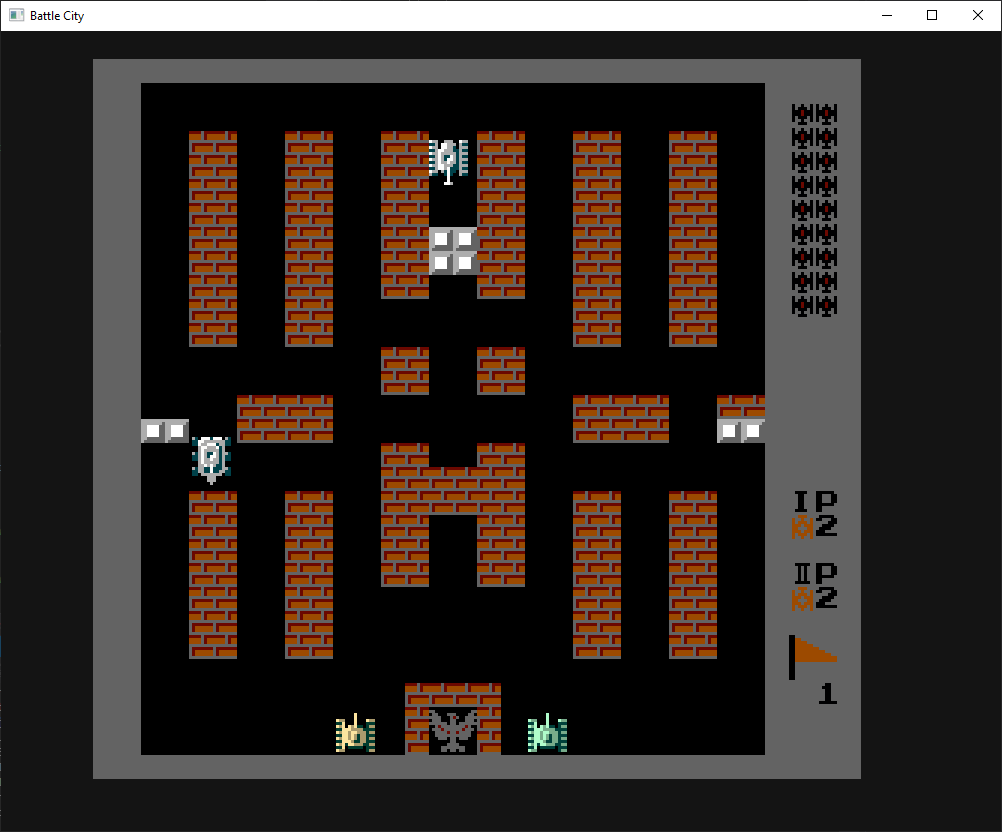
Перша та друга кнопки переводять гравця до самої гри, але з різною кількістю гравців. Третя переводить до таблиці рекордів. Четверта закриває гру.



6.2 Таблиця рекордів

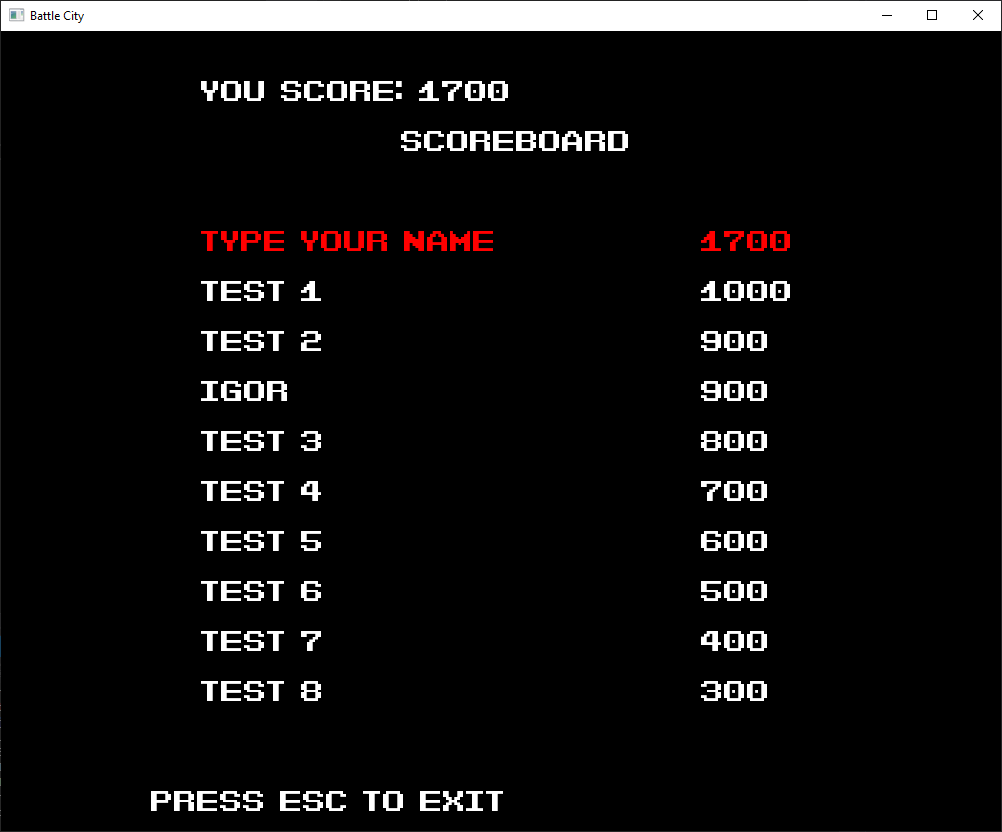


6.3 Сцена після натиснення 1 PLAYER



6.4 Сцена після натиснення 2 PLAYERS

Після завершення гри відкривається таблиця рекордів з можливістю ввести своє ім’я, якщо бали вищі за останнє місце



6.5 Таблиця рекордів з можливістю ввести ім’я

# 7. ІНСТРУКЦІЯ АДМІНІСТРАТОРА

Реалізація комп’ютерної гри «Battle City» була виконана в середовищі програмування Microsoft Visual Studio 2019 та на об’єктно-орієнтовній мові програмування C++.

Середовище програмування можна скачати з офіційного сайту <https://visualstudio.microsoft.com/ru/vs/>. Після встановлення та налаштування Слід відкрити головний файл проекту. З документацією можна ознайомитися за посиланням <https://docs.microsoft.com/ru-ru/visualstudio/windows/?view=vs-2019>.

Також знадобиться графічна бібліотека SFML яку можно дістати на офіційному сайті <https://www.sfml-dev.org/download.php>. Але вона вбудована в проект тому її потрібно завантажити лише в випадку зміни цільової платформи або для оновлення до новішої версії.

Ще використовується Dear ImGui та JSON for Modern C++ які можна знайти на їхніх сайтах <https://github.com/ocornut/imgui> та <https://github.com/nlohmann/json>. Вони не залежать від цільової платформи і тому їх потрібно завантажити тільки для оновлення.

Для покращення контролю версій проекта слід встановити систему контролю версій Git (https://git-scm.com/downloads).

Проект має репозиторій на сайці github.com та доступний за посиланням <https://github.com/XimikBoda/Battle_City>.

# 8. КЕРІВНИЦТВО КОРИСТУВАЧА

Комп’ютерна гра «Battle City» призначена для розваги та дружнього змагання осіб різного віку.

Для роботи програми необхідно мати 64-розрядну операційну систему Windows. Для запуску програми необхідно запустити файл Battle\_City.exe.

Для комфортного користування потрібно мати клавіатуру (керування відбувається за допомогою WASD та пробіл для першого гравця; стрілки та правий cntl для другого)

Основні можливості програми та їх застосування:

1. Вибір розміру вікна та можливість повноекранного режиму (по клавіші F11);
2. Для того щоб обрати рівень складності потрібно натискати в вправо чи в ліво в головному меню
3. Щоб переглянути таблицю рекордів треба натиснути на кнопку Records;
4. Для виходу із гри потрібно натиснути кнопку Exit.

Якщо після запуску виникне помилка про відсутній vcruntime140.dll потрібно встановити Розповсюджений пакет Visual C ++ для Visual Studio 2015 за посиланням <https://www.microsoft.com/ru-ru/download/details.aspx?id=48145> (потрібно обрати версію x64)

# ВИСНОВКИ

В результаті виконання курсової роботи було створено проект комп’ютерної гри «Battle City» у середовищі розробки Visual Studio. В ході роботи на практиці були засвоєні: навички використання основних принципів програмування на об’єктно-орієнтованих мовах програмування на прикладі мови C++.

Під час роботи над проектом були набуті навички роботи з графічною бібліотекою SFML та бібліотекою для парсингу даних JSON for Modern C++.

# ВИКОРИСТАНІ ДЖЕРЕЛА

1. Документація по C++ <https://docs.microsoft.com/en-us/cpp/?view=msvc-160>
2. Документація по SFML <https://www.sfml-dev.org/documentation/2.5.1/>
3. Документація по Dear ImGui <https://github.com/ocornut/imgui/wiki>
4. Документація по JSON for Modern C++ <https://github.com/nlohmann/json>
5. Робота з ресурсами windows <https://docs.microsoft.com/en-us/cpp/windows/resource-files-visual-studio?redirectedfrom=MSDN&view=msvc-160>
6. Опис деяких моментів реалізації оригинальної гри гри «Battle City» <https://habr.com/ru/post/142126/>
7. Основний опис оригинальної гри гри «Battle City» <https://ru.wikipedia.org/wiki/Battle_City>
8. Оригінальний бінарний образ комп'ютерної гри «Battle City» <http://dendy.com.ua/play/87>.
9. Шрифт подібний до оригінального <https://fonts2u.com/barcade-brawl-regular.font>
10. Спрайти для гри <https://www.deviantart.com/modelsandsprites/art/NES-Battle-City-General-Sprites-501889409>