

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ПОЛТАВСЬКИЙ ПОЛІТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Циклова комісія дисциплін програмної інженерії

КУРСОВА РОБОТА

з дисципліни: «Основи програмування та АМ» на тему
Морський бій

Виконав: здобувач освіти 2 курсу,
групи 25
спеціалізації «Розробка
програмного забезпечення»

Любченко О. В.

(прізвище та ініціали)

Керівник

Олійник В. В.

(прізвище та ініціали)

(підпис)

Полтава – 2023

ЗМІСТ

ВСТУП	3
1. ПОСТАНОВКА ЗАДАЧІ.....	4
1.1. Постановка задачі	4
1.2. Основні вимоги до програми.....	4
2. ОПИС ПРОЕКТНИХ РІШЕНЬ	5
2.1. Вибір мови програмування	5
2.4. Алгоритм програми	8
2.4. Тестування	11
3. ІНСТРУКЦІЯ КОРИСТУВАЧА	13
3.1. Початок	13
3.2. Розстановка власних кораблів	13
3.3. Розставлення кораблів комп'ютера.....	13
3.4. Процес гри	13
3.5. Вихід з гри	14
ВИСНОВКИ.....	15
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	16
ДОДАТОК А. ЛІСТИНГ ПРОГРАМИ	17

ВСТУП

«Морський бій» – це гра для двох гравців, де вони по черзі намагаються потопити кораблі супротивника, називаючи координати на його карті. Якщо вони влучають у правильне місце, то корабель або його частина «тоне», а гравець отримує право на наступний хід. Ціль гравця – першим потопити всі кораблі супротивника.

Вперше, як настільну гру, «Морський бій» випустила компанія Milton Bradley Company у 1931 році під іспанською назвою «Batalla Naval». Ця компанія також відома створенням таких відомих ігор, як «Candyland» та «Twister».

В наш час люди, зазвичай, грають в «Морський бій» на папері довго креслячи «поле бою» та малюючи на ньому кораблі. Настав час змінити це, за допомогою програмного продукту цієї курсової роботи ви зможете з легкістю грати в цю захоплюючу гру, не витрачаючи багато часу на створення потрібних малюнків. Все це за вас зможе зробити програма, миттєво створивши віртуальне поле бою, та дозволить вам швидко розставити на ньому кораблі.

Окрім веселого проведення часу ця гра також здатна розвивати наступні якості:

- логічне мислення;
- пам'ять;
- вміння аналізувати дії опонента;
- використовувати дані аналізу в наступних раундах.

Метою курсової роботи є відпрацювання навичок розробника, створюючи гру «Морський бій».

Завданням курсової роботи є побудова алгоритмів гри та створення програмного продукту на їхній основі.

1. ПОСТАНОВКА ЗАДАЧІ

1.1. Постановка задачі

Треба розробити комп'ютерну версію гри «Морський бій», яка забезпечить гравця механіками класичної версії:

- створення ігрових полів;
- розміщення кораблів;
- пострілів, влучань та промахів;
- пошкодження частин кораблів;
- оголошення переможця.

1.2. Основні вимоги до програми

Програмний продукт має бути виконаний мовою програмування C у середовищі Visual Studio Code за допомогою компілятора GCC.

- програма має створити віртуальні ігрові поля 10x10;
- надати можливість гравцю та комп'ютеру розставити по одному кораблю класів Aircraft Carrier (п'ятипалубний авіаносець), Battleship (чотирипалубний лінійний корабель), Cruiser (трипалубний крейсер), Destroyer (двопалубний ескадрений міноносець) та Submarine (однопалубний підводний човен);
- має бути забезпечена механіка пострілів по кораблях супротивника, а також система виявлення їхніх частин у обраних для влучання координатах, від якої буде надалі залежати статус координат;
- якщо всі кораблі одного з гравців знищено то програма має оголосити переможця та запропонувати зіграти знову;
- програма має бути консольним додатком.

2. ОПИС ПРОЕКТНИХ РІШЕНЬ

2.1. Вибір мови програмування

Мова програмування C є однією з найпоширеніших і впливових мов програмування загального призначення. Вона була розроблена у 1970-х роках Деннісом Річі в лабораторіях Bell Telephone Laboratories для розробки операційної системи Unix. З тих пір C стала однією з найпопулярніших мов програмування, застосовуваних у різних областях, включаючи системне програмування, розробку драйверів, вбудовані системи, графічні інтерфейси та багато іншого.

Основна особливість мови C полягає в її простому та лаконічному синтаксисі. Код на мові C організований у функції, які складаються з декларацій та операторів. Синтаксис мови C дозволяє легко описувати алгоритми та структури даних, забезпечуючи чіткість та зрозумілість коду.

Мова C надає розробникам низькорівневі можливості, зокрема пряму маніпуляцію з пам'яттю та адресами. Це дозволяє ефективно управляти ресурсами і писати швидкий та ефективний код. Компілятори мови C перетворюють програмний код у машинний код, що може бути виконаний на цільовій платформі.

Однією з переваг мови C є її портативність. Програми, написані на мові C, можуть бути перенесені з однієї платформи на іншу без змін у вихідному коді. Це робить мову C популярним вибором для розробки крос-платформових програм.

Мова C також володіє багатофункціональністю. Вона надає розробникам доступ до різноманітних бібліотек та функцій, що спрощують роботу з введенням/виведенням даних, роботою з рядками, математичними обчисленнями та іншими завданнями.

Мова C підтримує модульність, що дозволяє розбивати програму на незалежні функції та структури даних. Це полегшує читабельність коду, сприяє його супроводу та поліпшує перевикористання коду.

Крім того, мова C надає доступ до низькорівневих операцій, таких як робота з пам'яттю та бітові операції. Це дозволяє розробникам працювати на більш прямому рівні та досягати оптимальної продуктивності.

Мова C є основою для багатьох інших мов програмування, таких як C++, C#, Objective-C та інші. Вона має велику спільноту розробників та багато ресурсів та документації, що дозволяє швидко навчитися та ефективно використовувати цю мову.

2.2. Опис середовища розробки

Visual Studio Code (VS Code) є однією з найпопулярніших і потужних крос-платформових інтегрованих середовищ розробки (IDE). Розроблена компанією Microsoft, ця програма здобула широку популярність серед розробників усіх рівнів навичок та в різних галузях програмування.

Однією з головних переваг Visual Studio Code є його легкість використання та налаштування. Він має інтуїтивний та зручний інтерфейс, що дозволяє розробникам швидко освоїти його функціонал. Крім того, VS Code підтримує велику кількість розширень, які дозволяють налаштувати його під індивідуальні потреби кожного розробника.

VS Code надає підтримку для багатьох мов програмування, включаючи C, C++, Python, JavaScript, HTML, CSS та багато інших. Це робить його універсальним інструментом для розробки різних типів проектів. Крім того, VS Code підтримує інтеграцію з різними системами керування версіями, такими як Git, що спрощує спільну роботу над проектами в команді.

У Visual Studio Code також є багато корисних функцій, які полегшують розробку програм. Наприклад, він має вбудовану підсвітку синтаксису, автодоповнення коду, відладку, пошук та заміну тексту, вбудований термінал та багато інших інструментів, що допомагають розробникам ефективно працювати над своїми проектами.

Однією з важливих особливостей Visual Studio Code є його розширюваність. Він базується на архітектурі з плагінами, що дозволяє

розробникам створювати власні розширення та теми, розширюючи функціонал програми. Це дозволяє адаптувати VS Code до власних потреб та працювати з різними типами проектів і мов програмування.

Важливо зазначити, що Visual Studio Code є безкоштовною програмою з відкритим вихідним кодом, що робить його доступним для всіх розробників. Він підтримується активною спільнотою, що означає наявність великої кількості ресурсів, документації та підтримки, що сприяє швидкому розвитку та вдосконаленню програми.

Загалом, Visual Studio Code – це потужне та зручне середовище розробки, яке надає розробникам широкий набір інструментів для ефективної роботи над проектами різної складності. Його легкість використання, розширюваність та підтримка багатьох мов програмування роблять його популярним вибором серед програмістів по всьому світу.

2.3. Опис компілятора

GCC (GNU Compiler Collection) є одним з найпопулярніших і потужних компіляторів загального призначення, розробленим Фондом вільного програмного забезпечення (GNU). Він є відкритим і безкоштовним компілятором, доступним для різних платформ, включаючи Linux, Windows і macOS.

GCC підтримує багато мов програмування, включаючи C, C++, Objective-C, Fortran, Ada і деякі інші. Цей компілятор забезпечує можливість перетворення вихідного коду написаного на цих мовах у виконуваний машинний код або код проміжного рівня, який може бути виконаний віртуальною машиною.

GCC пропонує різні оптимізації коду, що дозволяє покращити продуктивність програм. Він використовує різноманітні техніки, такі як оптимізація розгортки циклів, оптимізація злиття ланцюгів, оптимізація з плаваючою комою та інші. Це допомагає розробникам створювати швидкі та ефективні програми.

GCC також має розширений набір інструментів і опцій компіляції. Він надає можливість контролювати процес компіляції, включаючи опції оптимізації, налаштування поведінки компілятора, використання специфічних функцій і бібліотек. Крім того, GCC підтримує різні архітектури процесорів і операційні системи, що робить його універсальним компілятором для різних платформ.

Окрім компіляції, GCC також надає можливості лінкування, статичної та динамічної збірки, використання бібліотек і розробку кернелів операційних систем. Це робить його потужним інструментом для розробки програмного забезпечення різного рівня складності та масштабу.

GCC є відкритим проектом з активною спільнотою розробників, що постійно вдосконалює компілятор і розширює його функціонал. Вона забезпечує документацію, підтримку і забезпечує наявність різноманітних ресурсів, які допомагають розробникам ефективно використовувати GCC.

Загалом, GCC є потужним і надійним компілятором, який дозволяє розробникам компілювати й оптимізувати код для різних мов програмування та платформ. Його широкий функціонал, розширюваність і активна спільнота роблять його популярним вибором для розробки програмного забезпечення.

2.4. Алгоритм програми

Після запуску програми вона ініціалізує головне меню, в якому гравець зможе або почати гру, або вийти з неї. У випадку, якщо гравець не бажає грати, програма завершує роботу, інакше запускає бій.

Щоб розпочати гру програма створює ігрові поля та запускає підпрограму «Розставлення кораблів гравця». Пізніше дію цієї механіки виконує комп'ютер зі своїм флотом.

Після цього відбувається процес самої гри із застосуванням інших її механік, який запускається з підпрограмою «Гра».

Після завершення бою програма пропонує гравцю зіграти ще раз, або ж вийти з програми.

Алгоритм роботи програми зображено блок-схемою (Рисунок 2.1).

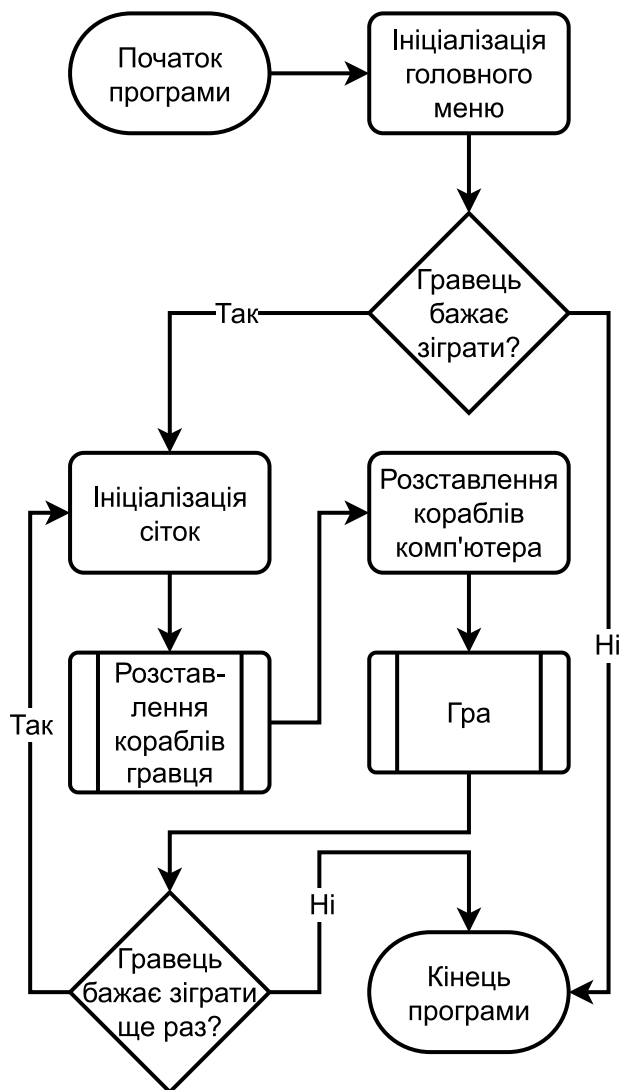


Рисунок 2.1 – Алгоритм роботи програми

При розставленні флоту гравця програма перевіряє який корабель буде поставлено наступним, якщо це початок гри, то ним буде Aircraft Carrier, після його встановлення буде оброблено Battleship, адже він на палубу менший, і так поки кораблі не закінчатся.

При розміщенні корабля програма запитує у гравця координати та орієнтацію (горизонтальна/вертикальна) для його встановлення.

Потім перевіряє, чи можна встановити корабель в заданому місці, і якщо воно вільне, то програма робить це і переходить до наступного.

Якщо ж всі кораблі розставлено, то підпрограма завершується.

Алгоритм роботи підпрограми «Розставлення кораблів» зображено блок-схемою (Рисунок 2.2).

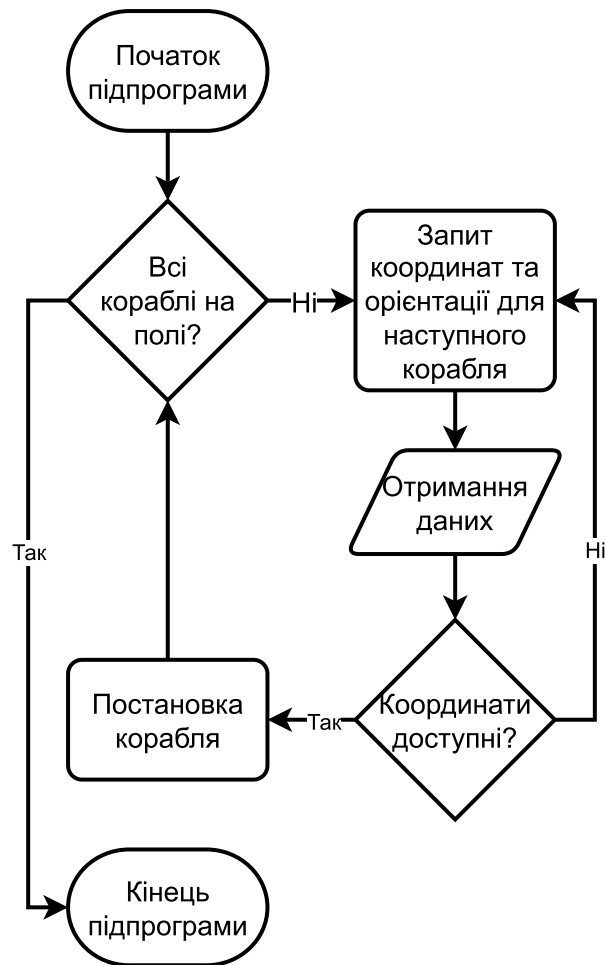


Рисунок 2.2 – Алгоритм роботи підпрограми «Розставлення кораблів»

Підпрограма «Гра» містить в собі виконання класичних механік гри «Морський бій».

Після початку підпрограми надається право пострілу гравцю. Програма запитує координати для атаки та виконує її. Після цього перевіряє чи було влучання в корабель супротивника. Якщо воно було, то програма перевіряє чи не скінчився флот в опонента. У випадку підтвердження цього статусу гра оголошує переможця, якщо ж кораблі ще є, то програма знову запитує у гравця координати для повторення пострілу. Якщо ж влучання не було, то програма надає право пострілу комп'ютеру.

Після чого повторює цю процедуру з опонентом, і так поки в когось не закінчаться кораблі.

Алгоритм роботи підпрограми «Гра» зображено блок-схемою (Рисунок 2.3).

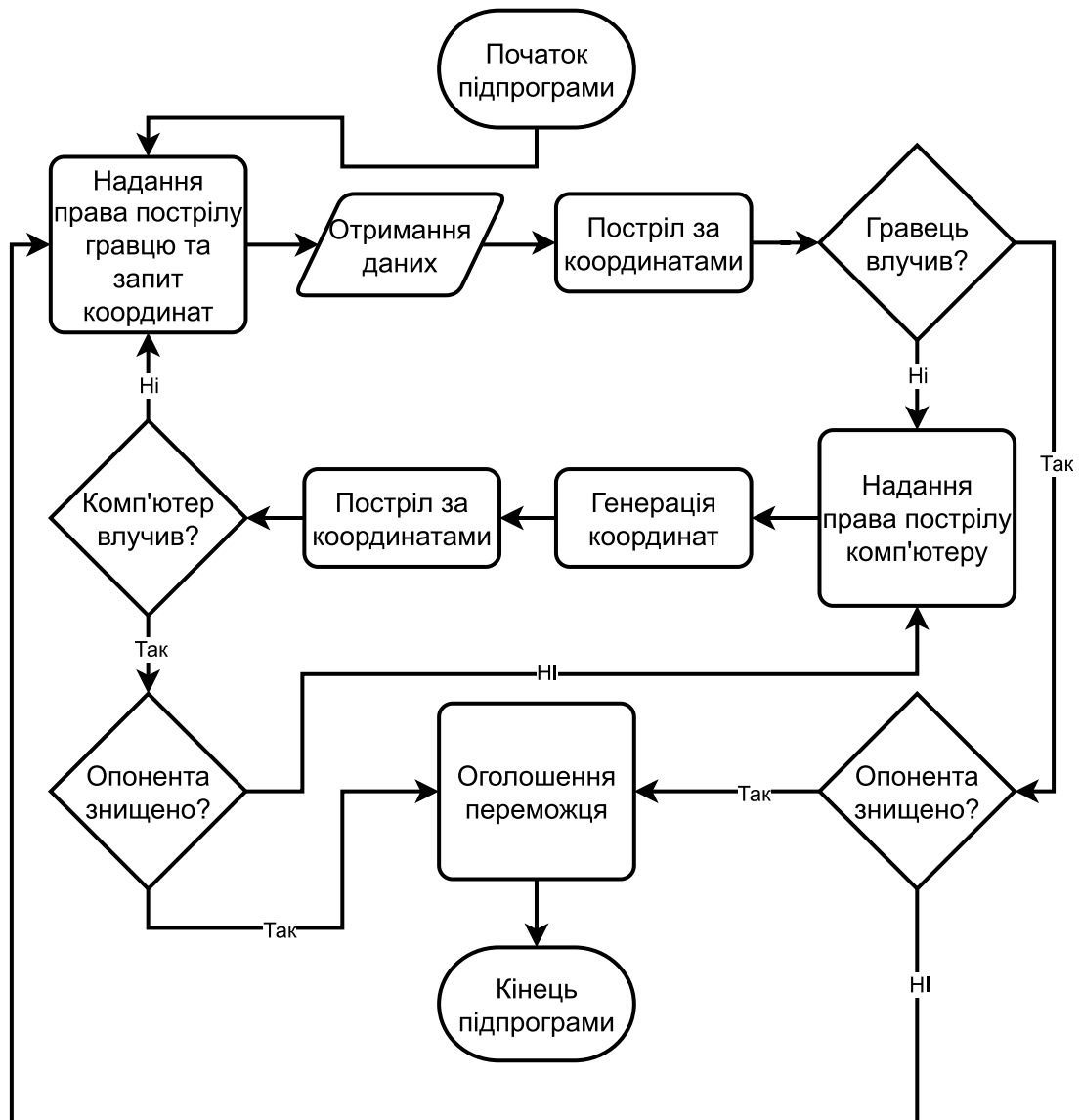


Рисунок 2.3 – Алгоритм роботи підпрограми «Гра»

2.4. Тестування

Для тестування було обрано методи чорної та білої скриньки, а також мавпячий тест.

Тестування чорної скриньки для програми – це процес перевірки функціональності програмного забезпечення без доступу до його внутрішньої

реалізації або структури коду. Під чорною скринькою розуміється програма, яка сприймає певні вхідні дані й повертає відповідні результати, але не відомо, як саме ці результати були отримані.

Метод тестування білої скриньки для програми, відомий також як метод функціонального тестування, зосереджується на внутрішніх аспектах програмного продукту. Цей метод вимагає доступу до вихідного коду програми та розуміння її внутрішньої структури.

Мавпячий тест – це процес, під час якого програма тестується шляхом випадкового генерування подій і введення випадкових даних. Це може виявити можливі проблеми в програмі, такі як збої, витоки пам'яті або некоректну обробку помилок.

Результат тестування чорної скриньки: програма має зрозумілий інтерфейс, тому для роботи з нею не потрібно знати її внутрішню реалізацію. Тестувальники зазначали що програма працює стабільно й адекватно реагує на випадково неправильні введення даних.

Результат тестування білої скриньки: якщо знати внутрішню реалізацію програми, то припускаєшся меншій кількості помилок під час введення даних, а в такому випадку програма працювала повністю справно та без нарікань.

Результат мавпячого тестування: програма оснащена рядом заходів, які будуть захищати нормальну роботу програми від неправильних введень. Тому програма не виходить з ладу під час постійного введення некоректних даних. Єдине, якщо ви спочатку ввели правильні дані, припустимо для атаки, а потім, через пробіл, ввели ще щось, то програма все одно пропустить вашу атаку, адже має властивість ігнорування даних, які не стосуються формату запитів.

3. ІНСТРУКЦІЯ КОРИСТУВАЧА

3.1. Початок

Після запуску програми ви побачите головне меню. Виберіть опцію «1. Start Game», щоб розпочати гру. Якщо ви бажаєте вийти з гри, виберіть опцію «2. Quit».

3.2. Розстановка власних кораблів

Після початку гри вам буде запропоновано розставити свої кораблі на ігровому полі. Ви побачите порожню сітку, позначену дефісами («-»), що представляє ваше поле. Вам буде поставлено запитання про позицію й орієнтацію кожного з кораблів, які ви повинні розмістити. Введіть координати (x, y) та орієнтацію («h» – горизонтально або «v» – вертикально) для кожного корабля. Якщо ви ввели щось неправильно, то вгорі ви побачите повідомлення «Invalid input. Try again.». У такому випадку ви повинні будете ввести дані знову. З розташуванням кожного корабля ви побачите оновлене поле з вашим флотом. А після розміщення останнього до цієї процедури перейде комп'ютер.

3.3. Розставлення кораблів комп'ютера

Після розміщення ваших кораблів комп'ютер автоматично розставить власні. Ви побачите повідомлення «Computer is placing it's ships.» під час цієї процедури.

3.4. Процес гри

Після розташування кораблів обох флотів гра розпочнеться. Гра проходить по черзі між вами та комп'ютером в залежності від результату ваших атак. Ваш хід розпочинається. Ви побачите поле комп'ютера у вигляді сітки. А також рахунок з кількістю цілих палуб кораблів, як ваших, так і комп'ютера. Чим менше значення на вашому рахунку, тим гірше ваше становище. Почніть атаку, вказавши координати (x, y) точки, яку ви хочете

атакувати. Якщо ви введете неправильні координати (наприклад, місце, яке ви вже атакували), вам вгорі буде показано повідомлення «Invalid move. Try again.». У такому випадку ви повинні ввести дані знову. Після вашого ходу ви побачите повідомлення «Acknowledged!», воно засвідчує те, що ви ввели правильні дані та почалася атака. В кінці атаки ви отримаєте одне з двох повідомлень: «Hit!» або «Miss!» вони позначають що ви влучили, або промахнулися відповідно. Якщо ви попали в корабель комп'ютера, координата буде позначена «Х» на його полі. Інакше координата буде позначена «О». Після вашого ходу право на нього передається комп'ютеру (у випадку, якщо ви промахнулися, інакше ви атакуєте ще раз). Комп'ютер автоматично обирає випадкову дійсну координату на вашому полі для атаки. Після ходу комп'ютера ви так само побачите повідомлення «Hit!» або «Miss!» залежно від результату ходу комп'ютера. Гра триває, поки всі палуби кораблів одного з гравців не будуть потоплені, а його рахунок не буде дорівнювати нулю. Якщо всі кораблі комп'ютера потоплені, ви побачите повідомлення «Victory!», вам покажуть ваше поле і гра закінчиться. Те ж саме відбудеться і якщо увесь ваш флот буде потоплено, але ви побачите повідомлення «Defeat!» і гра закінчиться, показавши ваше розбите поле. Якщо у вас є бажання зіграти ще раз, то після ініціалізації головного меню оберіть опцію «1. Start Game».

3.5. Вихід з гри

Після завершення гри ви можете вибрати опцію «2. Quit» у головному меню, щоб вийти з неї. Після виходу з гри ви побачите повідомлення «Exiting the game. Goodbye!» і програма завершить роботу.

ВИСНОВКИ

У цій курсовій роботі була розроблена консольна гра «Морський бій» з використанням мови програмування C. Процес розробки гри виконувався в середовищі Visual Studio Code з використанням компілятора GCC.

Гра "Морський бій" є класичною грою, де користувач має зіграти проти комп'ютера, розташовуючи свої кораблі на ігровому полі й намагаючись потопити всі кораблі супротивника. У цій роботі гра була реалізована у консольному інтерфейсі, де користувач може вводити координати атаки й бачити результат своїх ходів.

Для розробки гри були використані основні концепції мови програмування C, такі як змінні, умовні оператори, цикли та функції. Було створено функції для генерації ігрового поля, розташування кораблів, обробки ходу користувача та комп'ютера, перевірки на потоплення кораблів, а також для відображення стану гри на екрані.

Результатом роботи є повноцінна гра «Морський бій», яка може бути запущена у консолі. Гра має простий, але ефективний інтерфейс, що дозволяє користувачу зручно взаємодіяти з грою. Ігровий процес логічний і цікавий, завдяки чому гравець може насолоджуватися грою та розвивати свої тактичні навички.

Ця курсова робота дала можливість практично використати знання мови програмування C та навички розробки програм з використанням Visual Studio Code та компілятора GCC. Розробка гри «Морський бій» була відмінною можливістю для закріплення теоретичних знань та практичного використання програмних засобів.

В цілому, розробка консольної гри «Морський бій» у мові програмування C була цікавим та пізнавальним досвідом, що дозволило поєднати теоретичні знання з практичною реалізацією програмного продукту.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. МЕТОДИЧНІ ВКАЗІВКИ ДО ВИКОНАННЯ КУРСОВИХ РОБІТ ДЛЯ ЗДОБУВАЧІВ ОСВІТИ СПЕЦІАЛЬНОСТІ 121 «ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»: МЕТОДИЧНІ ВКАЗІВКИ / Відокремлений структурний підрозділ «Полтавський політехнічний фаховий коледж Національного технічного університету «Харківський політехнічний інститут»; [уклад.: В.В.Олійник]. – Полтава: ВСП ППФК НТУ «ХПІ», 2022. 22 с.
2. Татарчук, Д. Д. Програмування мовами С та С++ [Електронний ресурс] : навчальний посібник / Д. Д. Татарчук, Ю. В. Діденко; НТУУ «КПІ». – Електронні текстові дані (1 файл: 949,75 Кбайт). – Київ: НТУУ «КПІ», 2012. 112 с.
3. Miller D., Perry G. M. C Programming: Absolute Beginner's Guide. Que, 2014. 337 с.
4. Керніган Б. В., Річі Д. М. Мова програмування С, 1978. 343 с.
5. King K. N., C programming: A modern approach. New York : Norton, 1996. 661 с.
6. McGrath M. C++ Programming in Easy Steps (In Easy Steps). 3-тє вид. Computer Step, 2002. 192 с.
7. C docs – get started, tutorials, reference. Microsoft Learn: Build skills that open doors in your career. URL: <https://learn.microsoft.com/en-us/cpp/c-language/?view=msvc-170> (дата звернення: 15.06.2023).
8. Griffiths D., Griffiths D. Head First C. O'Reilly Media, Incorporated, 2012. 632 с.

ДОДАТОК А. ЛІСТИНГ ПРОГРАМИ

```
#include <stdio.h>

#include <stdlib.h>

#include <time.h>


#define GRID_SIZE 10

#define NUM_SHIPS 5


int main() {
    int choice;
    do {
        system("COLOR 03");
        printf("  < SEA BATTLE >\n");
        printf("-----\n");
        printf(" Main Menu:\n");
        printf("-----\n");
        printf(" 1. Start Game\n");
        printf(" 2. Quit\n");
        printf("-----\n");
        printf(" Enter your choice: ");
        scanf("%d", &choice);
        getchar();

        switch (choice) {
            case 1:
                system("cls");
                playGame();
                break;
            case 2:
                system("cls");
```

```
        printf(">>> Exiting the game. Goodbye!\n");  
        sleep(1);  
        break;  
    default:  
        system("cls");  
        printf(">>> Invalid choice. Try again.\n \n");  
  
    }  
} while (choice != 2);  
  
return 0;  
}
```

Повний код програми доступний на веб-сервісі GitHub за посиланням, що зображене на рисунку А.1.



Рисунок А.1 – Посилання на репозиторій