

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інженерії програмного забезпечення

Пояснювальна записка

до магістерської кваліфікаційної роботи
на ступінь вищої освіти магістр

на тему: **«РОЗРОБКА АЛГОРИТМУ ПІДБОРУ ПРОТИВНИКА ДЛЯ
МУЛЬТИМЕДІЙНОЇ ОНЛАЙН ГРИ ЖАНРУ MMORPG»**

Виконав: студент 6 курсу, групи ПДм-62

спеціальності 121 Інженерія програмного
забезпечення

(шифр і назва спеціальності)

Хом'як Д. М.

(прізвище та ініціали)

Керівник

Негоденко О. В.

(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

Нормоконтроль

(прізвище та ініціали)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення
Ступінь вищої освіти - «Магістр»
Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ
Завідувач кафедри
Інженерії програмного забезпечення
Негоденко О.В.
« » 2022 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Хом'як Дмитро Миколайович

1. Тема роботи: «Розробка алгоритму підбору противника для мультимедійної онлайн гри жанру MMORPG»

Керівник роботи Негоденко Олена Василівна, доцент кафедри, затверджені наказом вищого навчального закладу від — «12» жовтня 2022 року №122.

2. Строк подання студентом роботи 31.12.2022

3. Вхідні дані до роботи: Матеріали переддипломної практики, методи підбору, покращений рейтинговий підбір противника як основний алгоритм підбору в онлайн іграх, науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).

4.1. Поняття алгоритмів

4.2. Алгоритм рейтингового підбору противника

4.3. Покращення рейтингового алгоритму підбору як основа підбору противника

4.4. Застосування покращеного рейтингового алгоритму підбору

5. Перелік графічного матеріалу (презентація)

5.1. Мета, об'єкт та предмет дослідження

5.2. Алгоритм покращеного рейтингового підбору противника

5.3. Модель підбору противника

5.4. Модель персонажа та його предметів

6. Дата видачі завдання «14» жовтня 2022 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Отримання завдання на магістерську роботу	14.10.2022	
2	Огляд теоретичних засад алгоритмів підбору	17.10.2022	
3	Аналіз методів та засобів підбору противника в онлайн іграх	20.11.2022	
4	Розробка методів	22.11.2022	
5	Розробка моделей	23.11.2022	
6	Тестування моделей	24.11.2022	
7	Моделювання та аналіз результатів	26.11.2022	
8	Написання та оформлення пояснювальної записки	27.11.2022	
9	Розробка графічних та презентаційних матеріалів	11.12.2022	
10	Захист магістерської роботи	17.01.2023	

Студент _____ Хом'як Д. М.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Негоденко О. В.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина магістерської роботи 83 сторінки.

Ключові слова: АГОРИТМ ПІДБОРУ, ПЕРСОНАЖ, ПРОТИВНИК, МОДЕЛЬ ПЕРСОНАЖА, МОДЕЛЬ РЮКЗАКА, БОТ.

Об'єкт дослідження – процес підбору противника.

Предмет дослідження – алгоритми та методи рейтингового підбору противника для гри.

Мета роботи – покращення алгоритму підбору противника в мультимедійній онлайн грі.

Методи дослідження – теоретичні, аналіз науково-методичної літератури з теми застосування параметрів у розробці алгоритму підбору, розробка алгоритму підбору та загальні положення про алгоритми; емпіричні, проведення апробації проведених досліджень з теми розробки алгоритму підбору.

У дипломній роботі проаналізовано ігрову індустрію, описано декілька жанрів комп'ютерних ігор та їх особливості. Також, було проаналізовано найпопулярніші сучасні ігри, завдяки чому зрозуміло на яких алгоритмах вони працюють. До цього ж, були проаналізовані теоретичні засади алгоритму підбору противника, а саме: поняття «Покращений алгоритм підбору», його переваги та використання алгоритмів підбору. Було розглянуто багато можливих факторів, які могли б вплинути на підбір противника в бою, визначено основні параметри, які доступні в більшості MMORPG іграх, для того щоб описати принцип роботи алгоритму використовуючи тільки основні вхідні параметри, адже кожна гра унікальна та у кожному випадку потрібно модифікувати сам процес підбору. Також описано особливості застосування покращеного алгоритму підбору противника в мультимедійній онлайн грі.

Проведено апробацію розробленого алгоритму. Описана його реалізація та тестування з порівнянням.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	9
ВСТУП	10
1 ТЕОРЕТИЧНІ ЗАСАДИ АЛГОРИТМІВ, АНАЛІЗ ОНЛАЙН ІГОР ТА ЇХ АЛГОРИТМІВ	12
1.1 Розвиток ігрової індустрії	12
1.2 Огляд жанру MMORPG	16
1.3 Аналіз операційних систем для ігор	18
1.4 Аналіз ринку ігор	21
1.4.1 DOTA 2	21
1.4.2 Counter-Strike: Global Offensive	23
1.4.3 PUBG: BATTLEGROUNDS	25
1.5 Поняття алгоритму	27
1.6 Алгоритми підбору	30
1.7 Аналіз алгоритму підбору противника	32
1.7.1 Рейтингова система підбору гравця	32
1.7.2 Математичний опис алгоритму	33
2 ПРОЕКТУВАННЯ ТА РОЗРОБКА АЛГОРИТМУ ПІДБОРУ ПРОТИВНИКА	35
2.1 Концепція та сценарій алгоритму	35
2.2 Підготовка елементів алгоритму	36
2.2.1 Модель персонажа та його сила	36
2.2.2 Модель рюкзака та його сила	41
2.2.3 Коефіцієнт перемог та поразок	44

2.3 Внутрішня ігрова логіка	46
3 АПРОБАЦІЯ РЕЗУЛЬТАТІВ РОБОТИ	49
3.1 Реалізація створеного алгоритму	49
3.1.1 Опис гравців та режимів гри	49
3.1.2 Характеристики підбору противника	52
3.1.3 Коефіцієнт підбору гравця.....	54
3.2 Результати застосування алгоритму	56
ВИСНОВКИ.....	63
ПЕРЕЛІК ПОСИЛАНЬ	64
Додаток А.....	66
Додаток Б	77

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ПЗ – програмне забезпечення

СУБД – Система Управління Базами Даних

IDE – Інтегроване середовище розробки

БД – База даних

PHP – Hypertext Preprocessor

API – Application Programming Interface

MMORPG – Massively Multiplayer Online Role-Playing Game

CS:GO – Counter-Strike: Global Offensive

MM – Match Making

PUBG – PlayerUnknown's Battlegrounds

ОС – Операційна система

PvP – Player versus player

PvE – Player versus Environment

NPC – Non-Player Character

ВСТУП

Кожного дня люди зіштовхуються з різного роду алгоритмами. Вони існували задовго до того як з'явилися перші комп'ютери. Потрібно отримати якусь хімічну сполуку? Зроби певну послідовність дій з певними реагентами. З цього випливає, що алгоритм – це виконання певної послідовності дій над чимось для отримання потрібного результату [5]. Більш доступними словами можна сказати «рецепт». Взагалі, поняття алгоритму зародилось в математиці. Саме вона почала вивчати дане поняття.

Кожен комп'ютер, та і взагалі любий сервіс, програма чи гаджет працює за заданим алгоритмом. Тобто в ньому прописаний сценарій, що потрібно зробити, щоб отримати заданий результат. Навіть більше, всі алгоритми працюють в різноманітних комбінаціях, завдяки чому можна виконувати найскладніші задачі. Може бути таке, що для того, щоб виконати якусь задачу, комп'ютер використовує більше сотні різних, але пов'язаних між собою алгоритмів, які в купі можна об'єднати в один великий.

Візьмемо за приклад алгоритм підбору. Його основна задача отримати якісь параметри, обробити їх, тобто порахувати та видати кінцевий результат, який може бути як підібраний якийсь товар, так і числовим значенням для наступного алгоритму.

Останні 5 років інформаційні технології розвиваються з великою швидкістю в усіх напрямках. Але особливо потужним розвитком можна характеризувати саму ігрову індустрію. У 2020 році світовий ринок ігор зріс на 23,1% — це найвищий показник за останнє десятиліття. Тому особливу увагу слід приділити саме цьому напрямку.

На даний момент існує безліч онлайн ігор, в яких є багато проблем з підбором противника для змагання. Якщо проаналізувати відгуки найпопулярніших онлайн ігор, то можна зрозуміти, що гравці постійно жаліються на систему підбору. Це свідчить про поганий або недороблений алгоритм підбору противника.

Актуальність обраної теми. Актуальність дипломної роботи полягає в тому, що є безліч різноманітних онлайн ігор, в яких неправильно або погано реалізований алгоритм підбору противника, через що виникає безліч негативних відгуків, складається погане враження у гравців, рейтинги гри падають і компанія, яка її розробила, втрачає прибуток. Як наслідок, компанії розпадаються, розробники втрачають роботу, у гравців складається погана думка про розробників та компанію в цілому.

Саме тому слід постійно покращувати, допрацьовувати, вводити додаткові параметри в алгоритм підбору противника.

Мета роботи – покращення алгоритму підбору противника в мультимедійній онлайн грі.

Об'єкт дослідження – процес підбору противника.

Предмет дослідження – алгоритми та методи рейтингового підбору противника для гри.

Щоб досягнути поставленої мети, слід виконати такі завдання:

1. Розглянути та опрацювати матеріали з обраної галузі;
2. Проаналізувати наскільки ефективно використання алгоритмів підбору в інших напрямках;
3. Обрати технології для розробки та програмні засоби реалізації;
4. Розробити алгоритм підбору противника для мультимедійної онлайн гри в жанрі MMORPG;
5. Провести апробації розробленого алгоритму та узагальнити результати дослідження.

Під час написання роботи було використано такі *методи дослідження*:

- теоретичні: аналіз науково-методичної літератури з теми використання алгоритмів підбору, розробка алгоритму підбору противника;
- емпіричні: проведення апробації досліджень з теми розробки алгоритму підбору противника.

1 ТЕОРЕТИЧНІ ЗАСАДИ АЛГОРИТМІВ, АНАЛІЗ ОНЛАЙН ІГОР ТА ЇХ АЛГОРИТМІВ

1.1 Розвиток ігрової індустрії

Ігрова індустрія зазвичай розділяють на кілька етапів розвитку.

1. Початковий етап (1970 – 1979) – розвиток перших відеоігор, використовуючи перші консолі, такі як Pong, Atari 2600 і Intellivision.

2. Ера IT-епохи (1980 – 1989) – ігрові консолі стали популярнішими, нові графічні технології і більш продумані ігри появилися на ринку, такі як Nintendo Entertainment System і Sega Genesis.

3. Ера 3D (1990 – 1999) – розвиток 3D-графіки і високоякісної анімації, запуск Sony PlayStation і Sega Saturn, з'явилися нові жанри ігор, такі як шутери від першої особи і рольові ігри.

4. Ера онлайн ігор (2000 – 2009) – з'явилися онлайн ігри, здатність грати з багатьма реальними гравцями без затримки, появилися мобільні ігри та гравці почали використовувати соціальні мережі для гри.

5. Ера мобільних ігор і консолей сучасного покоління (2010 – нині) – поява мобільних ігрових платформ, які дозволяють грати в ігри на смартфонах і планшетах, поява нового покоління консолей. Таких як PS4 і Xbox One, значний розвиток виробництва інді-ігор, зростання популярності онлайн ігор, таких як Fortnite, PUBG, League of Legends та інших.

За останні роки індустрія вирішила нові задачі, такі як використання машинного навчання у іграх, розвиток віртуальної реальності та інші. Це дає можливість створювати більш реалістичні і захоплюючі ігрові світи, що залучає більшу кількість гравців і збільшує популярність ігрової індустрії.

Розвиток ігрової індустрії також супроводжується ростом глобального ринку ігор. Якщо взяти світову економіку, то ігрова індустрія становить досить великий сектор економіки, і в більшості країн вона стає все більш популярною і важливою.

Розвиток інтернету також відіграв важливу роль у розвитку ігрової індустрії, завдяки цьому гравці можуть грати разом з іншими гравцями з усього світу, скачувати нові ігри, оновлення і додаткові матеріали, також можуть спілкуватись з командою та своєю спільнотою гри [20].

В майбутньому, ігрова індустрія має продовжувати розвиватися, завдяки новим технологіям і змінам поведінки споживачів. Очікується, що ринок мобільних ігор буде продовжувати рости, а виробники будуть використовувати машинне навчання і віртуальну реальність, щоб створювати ще більш реалістичні і захоплюючі ігрові світи. Онлайн ігри з великою кількістю гравців будуть продовжувати домінувати на ринку. Окрім цього, виробники будуть намагатися запровадити нові форми монетизації, такі як підписки та мікро-платежі [14].

В цілому, розвиток ігрової індустрії був і продовжує бути дуже швидким і динамічним, що змінює способи та відчуття гри. Різноманіття жанрів, платформ і способів гри зростає і дозволяє гравцям знаходити щось цікаве для себе.

Розглянемо загальний прибуток від ігор у всьому світі за останні роки та прогноз на майбутнє. Для цього використаємо статистику, яку надав сервіс Google разом з Newzoo. На рисунку 1.1 зображено тенденцію росту доходу від ігрової індустрії [1].



Рисунок 1.1 – Тенденція росту доходу від ігрової індустрії [1].

Також, розглянемо статистику кількості гравців по регіонах за весь 2021 рік. Як зображено на рисунку, загальна кількість сягає близько трьох мільярдів людей, а тенденція зростання зберігається вже не один рік. На рисунку 1.2 зображений графік кількості гравців по регіонах [1].

Опираючись на статистику, можна сказати, що майже 180 мільйонів гравців додалось за весь 2021 рік. В це число також входять люди, які повернулись в ігрову індустрію після довготривалого періоду. Цікаво, що більша половина всіх гравців – це жінки. Їх кількість складає 53% від загальної. Також, абсолютна більшість, а саме 73% людей витрачають витрачали коли-небудь будь-які кошти на ігри. Можна відзначити нові тенденції на світовому ігровому ринку:

1. Поява підписок та готовністю людей оплачувати їх.
2. Метавсесвіт, доповнена реальність та соціальні платформи, які зароджуються безпосередньо в самих іграх.

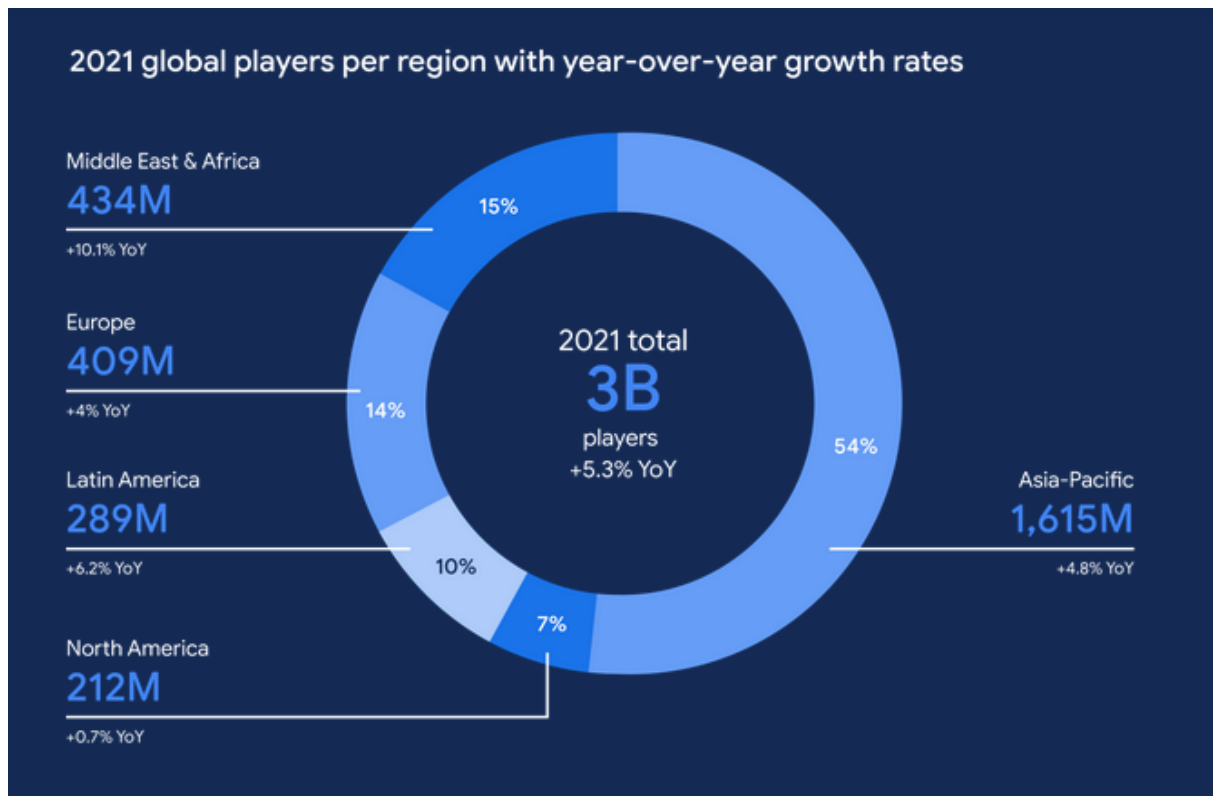


Рисунок 1.2 – Графік кількості гравців по регіонам [1]

За допомогою цієї ж статистики, можемо розглянути демографію в ігровій індустрії України. На рисунку 1.3 зображений демографічний графік в регіоні Україна [1].

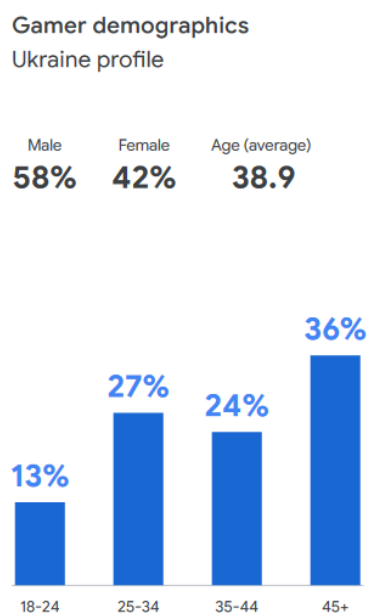


Рисунок 1.3 – Демографічний графік в регіоні Україна [1]

1.2 Огляд жанру MMORPG

MMORPG (Massively Multiplayer Online Role-Playing Game) є жанром відео ігор, які дозволяють гравцям виступати у ролі персонажа і взаємодіяти з іншими гравцями у світі, де грає одночасно багато гравців.

В таких іграх гравці можуть створювати свої власні персонажі, надавати їм різні властивості і навички, а також відкривати і досліджувати велику кількість локацій. В цих іграх також часто можна приєднатися до груп із іншими гравцями для виконання спільних задач, таких як бої проти босів, зачистку загадкових печер, чи спільних завдань.

Ігри в жанрі MMORPG зазвичай пропонують великий вибір квестів, що дозволяють гравцям відкривати нові локації і виконувати різні задачі, які допоможуть їм покращувати свого персонажа та отримувати нові ресурси. Вони також можуть брати участь у різноманітних подіях, які проводяться на сервері і дозволяють гравцям отримувати спеціальні нагороди і предмети [17].

В цілому, MMORPG забезпечує гравцям великий функціонал і велику кількість можливостей для відкриття і взаємодії в реалістичному віртуальному світі, де відповідальність і співпраця є ключовими елементами. MMORPG забезпечують гравцям широкі можливості для самореалізації, створення своєї власної історії, а також для навчання і покращення комунікаційних навичок в межах віртуальної групи.

Крім того, в MMORPG іграх можна обмінюватись, продавати та купувати предмети у інших гравців на віртуальному ринку, що дозволяє гравцям заробляти віртуальні гроші, які можуть бути використані для покупки потрібних предметів або спорядження.

Так, MMORPG ігри мають багато переваг, завдяки яким велика кількість гравців постійно в них грають. Вони популярні не лише завдяки можливості взаємодії з іншими гравцями, але й завдяки контенту, який стає доступним в ході гри, а також доступність відкритих світів, які дозволяють гравцям вільно досліджувати світ та виконувати різні завдання.

В MMORPG іграх гравці часто можуть об'єднуватися в групи, для спільної мети і виконання спільних завдань, що дозволяє збільшити соціальну складаючи гри. Ігрові сервери в MMORPG зазвичай надають гравцям доступ до великої кількості додаткового контенту, такого як нові задачі, події, предмети, зовнішній вигляд та інше.

В цілому, MMORPG ігри надають гравцям заворожливий світ для відкриття і взаємодії, де гравці можуть створювати свої власні персонажі, виконувати завдання, співпрацювати з іншими гравцями, досягати високих рівнів. Вони також дозволяють гравцям відчувати себе частиною спільноти з людьми з усього світу, навіть якщо вони фізично не зустрічаються.

Також важливо зазначити, що жанр MMORPG в залежності від ігри можуть бути різної складності завдання і вимоги для їх виконання. Таким чином, деякі можуть бути більш складні і потребувати більшої кількості часу та зусиль, а деякі бути більш доступними і не вимагати цього всього.

На рисунку 1.4 зображено приклад мультимедійної онлайн гри в жанрі MMORPG.



Рисунок 1.4 – Приклад мультимедійної онлайн гри в жанрі MMORPG

1.3 Аналіз операційних систем для ігор

Операційні системи, які найкраще підходять для комп'ютерних ігор, залежать від бажаної платформи та вимог до гри.

Windows є найпоширенішою операційною системою для ПК, і більшість комп'ютерних ігор розробляються спеціально для неї. Це дає велику кількість опцій для гравців, але не завжди гарантує найкращу роботу гри на комп'ютерах з меншою потужністю.

macOS є ідеальним вибором для гравців, які мають Apple-комп'ютери, але кількість ігор, які доступні на macOS значно менша, ніж на Windows.

Linux є безкоштовною операційною системою, і є деякі гри, які працюють на ній, але кількість доступних ігор для Linux також значно менша, ніж на Windows та macOS.

На рисунку 1.5 зображено операційну систему Windows.



Рисунок 1.5 – Операційна система Windows

На консолях таких як PlayStation, Xbox, Nintendo Switch теж можуть запускатися ігри, але в такому випадку ігор буде ще менше, не всі ігри,

розроблені для ПК, можуть запускатися на консолях. При порівнянні консолей між собою, важливо враховувати вимоги до гри, а також власні потреби гравця.

На рисунку 1.6 зображено операційну систему Linux.

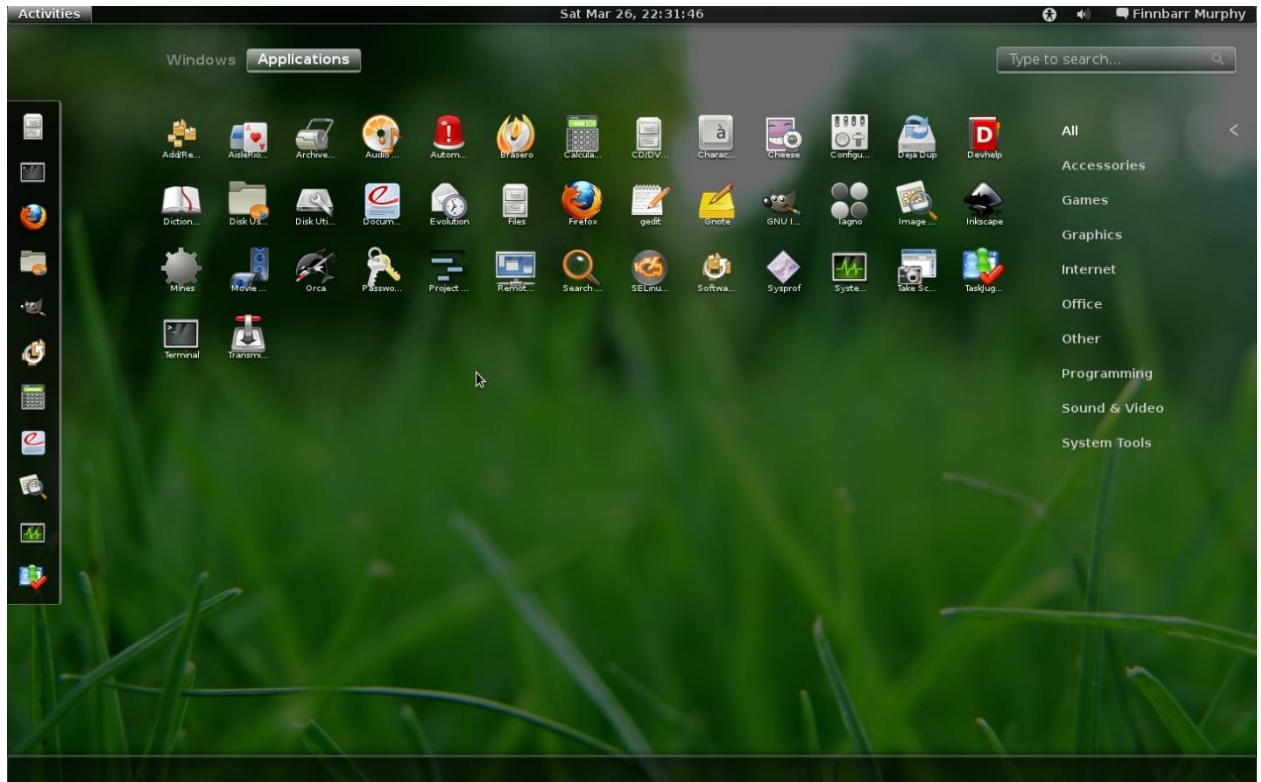


Рисунок 1.6 – Операційна система Linux.

Якщо говорити про ПК, то при виборі операційної системи для комп'ютерних ігор, слід враховувати наступні фактори:

1. Доступність ігор: Якщо ви плануєте грати в певні ігри, слід переконатися, що вони доступні на операційній системі, яку ви обираєте.
2. Продуктивність: Різні операційні системи можуть по-різному впливати на роботу гри. Якщо ви маєте менш потужний комп'ютер, то варто обирати операційну систему, яка займає менше ресурсів.
3. Підтримка: Деякі операційні системи можуть бути простішими в налаштуванні або встановленні оновлень для гри. Важливо переконатися, що

операційна система, яку ви обираєте, має достатню підтримку та документацію.

В цілому, вибір операційної системи для комп'ютерних ігор може бути важливим фактором для отримання найкращої роботи гри. Якщо ви плануєте грати в більшість ігор, розроблених для Windows і маєте потужний ПК, то найкращим варіантом буде Windows. Але якщо ви маєте обмежений список ігор і хвилюєтесь про надійність і безпеку, можливо Linux буде кращим варіантом. Порівняння та аналіз різних операційних систем може допомогти визначити, яка операційна система найкраще відповідає вашим потребам та вимогам для комп'ютерних ігор.

На рисунку 1.7 зображено операційну систему консолі PS5.

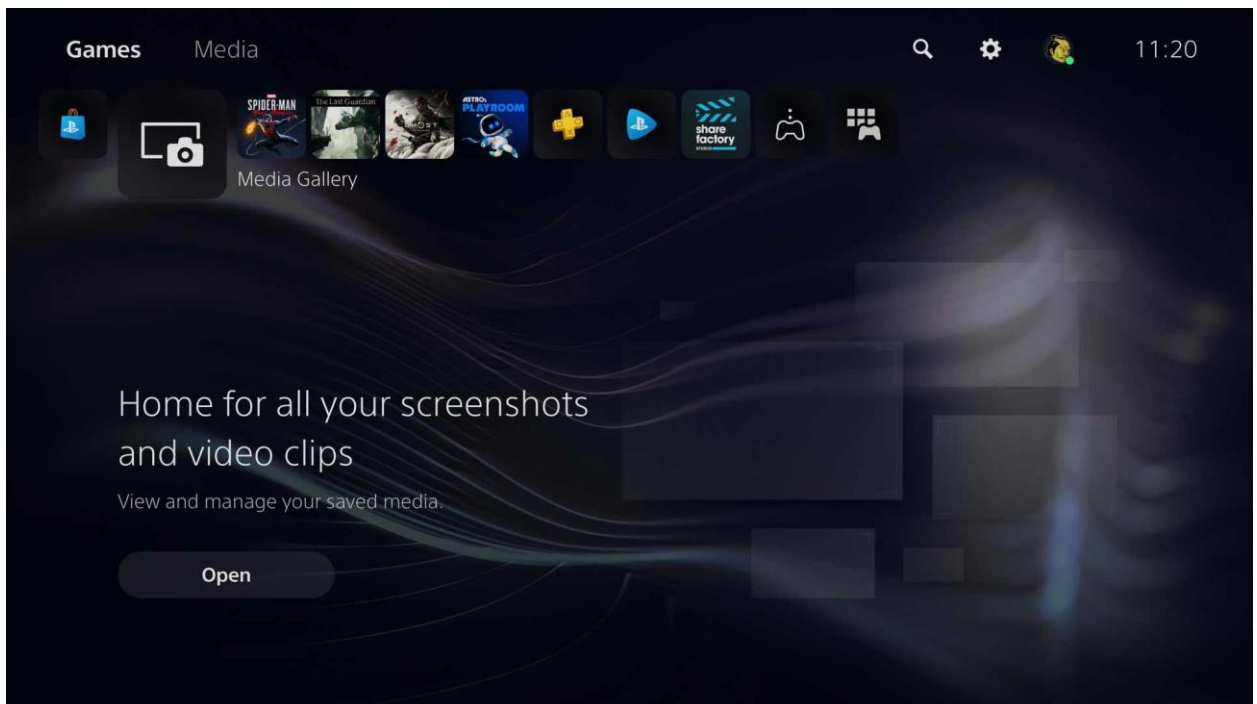


Рисунок 1.7 – Операційна система консолі PS5.

Звернувшись до досвіду гравців і експертів в галузі, можна зробити деякі заключення. Windows є найбільш поширеною операційною системою для ПК, саме тому більшість ігор розробляють з підтримкою саме цієї операційної системи, тому вибір ігор для Windows є найбільшим і буде краще відповідати

вимогам і потребам гравця в порівнянні з іншими ОС. В разі, якщо у вас є менш потужний комп'ютер або ви плануєте грати в обмежену кількість ігор, варто розглянути вибір Linux, яка займає менше ресурсів, але вибір ігор для неї буде обмеженим.

Якщо говорити про консолі, то вибір конкретної консолі залежить від вимог до гри та ваших персональних вподобань у дизайні та керуванні.

В цілому, вибір операційної системи для комп'ютерних ігор може важко, особливо якщо ви не знайомі з різними опціями. Для того щоб оптимізувати гру та поліпшити ваш досвід, рекомендується виконати детальний аналіз різних операційних систем та визначити яка з них найкраще відповідає вашим потребам та вимогам.

1.4 Аналіз ринку ігор

1.4.1 DOTA 2

В світі онлайн ігор однією з найпопулярніших є саме DOTA 2. Це багатокористувацька онлайн гра в жанрі MMORPG [2]. Алгоритми, які в ній використовуються, можна дійсно вважати одними з найскладніших і незрозумілих для всіх гравців. Проаналізувавши відгуки, можна відзначити, що в грі є рейтингова система підбору противника, яка на перший погляд може виявитись зрозумілою та простою, а з іншого саме вона є об'єктом більшості поганих відгуків. На рисунку 1.8 можна побачити приклад гри та її геймплей.



Рисунок 1.8 – Приклад гри в DOTA 2

Також, розглянемо рейтингову систему в даній грі. Завдяки чому в майбутньому можна буде визначити всі плюси та мінуси рейтингової системи DOTA 2.

Оскільки в цю гру одночасно грає близько одного мільйона гравців, то алгоритми, які в ній використовуються, мабуть бути приближені до ідеальних, адже щоб розділити таку велику кількість гравців потрібно добре постаратись.

Скачавши та погравши в дану гру, можна сказати, що гравці поділені на континенти, завдяки чому у людей не виникає проблем із великою затримкою.

Також, в DOTA 2, в якості рейтингової системи використовується MMR. Більш детально про неї описано нижче. Вона є однією з основних рейтингових систем взагалі, саме тому її алгоритми будуть взяті за приклад.

Окрім числового значення MMR, гравцям видають свої звання та красиву іконку, які напряду залежать від числа. Завдяки цьому можна обмежити пошук гравців між собою на рівні плюс або мінус одне звання.

На рисунку 1.9 зображено звання та відповідний рейтинг для його отримання.

Statistics

New ranked matchmaking system

	 Herald	 Guardian	 Crusader	 Archon	 Legend	 Ancient	 Divine
	0	840	1680	2520	3360	4200	5040
★	140	980	1820	2660	3500	4340	5180
★★	280	1120	1960	2800	3640	4480	5320
★★★	420	1260	2100	2940	3780	4620	5460
★★★★	560	1400	2240	3080	3920	4760	5600
★★★★★	700	1540	2380	3220	4060	4900	5740

Рисунок 1.9 – Система рейтингу гравців в DOTA 2

1.4.2 Counter-Strike: Global Offensive

Наступною за популярністю грою є Counter-Strike: Global Offensive. Це багатокористувацька комп'ютерна гра в жанрі шутер від першої камери [3].

Гра має безліч різноманітних режимів, які між собою зовсім не схожі, але суть гри при цьому залишається єдиною. Перемогти своїх суперників. Режим, де використовується алгоритм підбору, називають Match Making або скорочено ММ.

На рисунку 1.10 зображений приклад режиму ММ в грі CS:GO

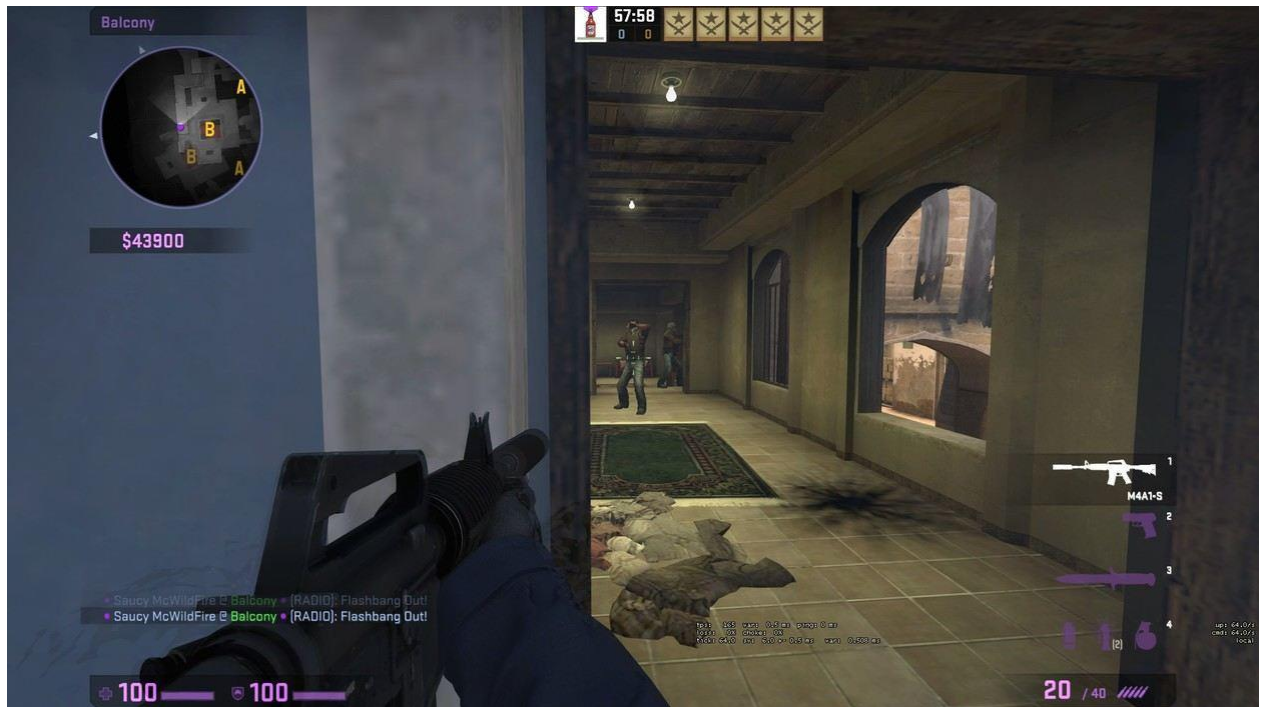


Рисунок 1.10 – Приклад режиму ММ в грі CS:GO

Проаналізувавши даний режим, виявилось, що в ньому є свій аналог рейтингової системи підбору, який працює за схожим алгоритмом.

В CS:GO також є звання та рейтинг. Але кількість необхідних балів для отримання кожного звання приховане, як і кількість втрачених та отриманих балів за поразку або перемогу в матчі.

На рисунку 1.11 зображені звання рейтингової системи підбору гравця в грі CS:GO.



Рисунок 1.11 – Звання рейтингової системи підбору гравця в грі CS:GO

Пошук противника здійснюється порівнюючи саме звання, а не числове значення. Якщо в DOTA 2 це було плюс або мінус одне звання при пошуку, то в CS:GO це може бути аж цілих чотири пункти в більшу або меншу сторону.

Оскільки, кількість рейтингових балів подивитись не можливо, відповідно і визначити кількість отриманих балів за перемогу чи втрачених за поразку також.

1.4.3 PUBG: BATTLEGROUNDS

Наступною за популярністю і останньою грою буде PUBG. PUBG: BATTLEGROUNDS – це найпопулярніша багатокористувацька та безкоштовна онлайн гра в жанрі шутер, або ж королівської битви [4].

Вона стала популярною саме завдяки своєму режиму королівської битви. Сто гравців змагаються між собою до останнього. Головною задачею

гри є вижити до кінця змагання та будь-яким чином позбутися своїх противників.

На рисунку 1.12 зображений приклад режиму королівської битви в грі PUBG.



Рисунок 1.12 – Приклад режиму королівської битви в грі PUBG

В грі також присутня рейтингова система, завдяки якій найкращі гравці можуть змагатися між собою. Вона складається з шести рівнів, які також поділені на п'ять підрівнів. На кількість отриманих балів після битви впливає багато чинників. Такі як кількість допомоги при вбивстві противника, кількість вбитих противників, місце, яке зайняв гравець та багато інших.

На рисунку 1.13 зображені рівні рейтингової системи в грі PUBG.

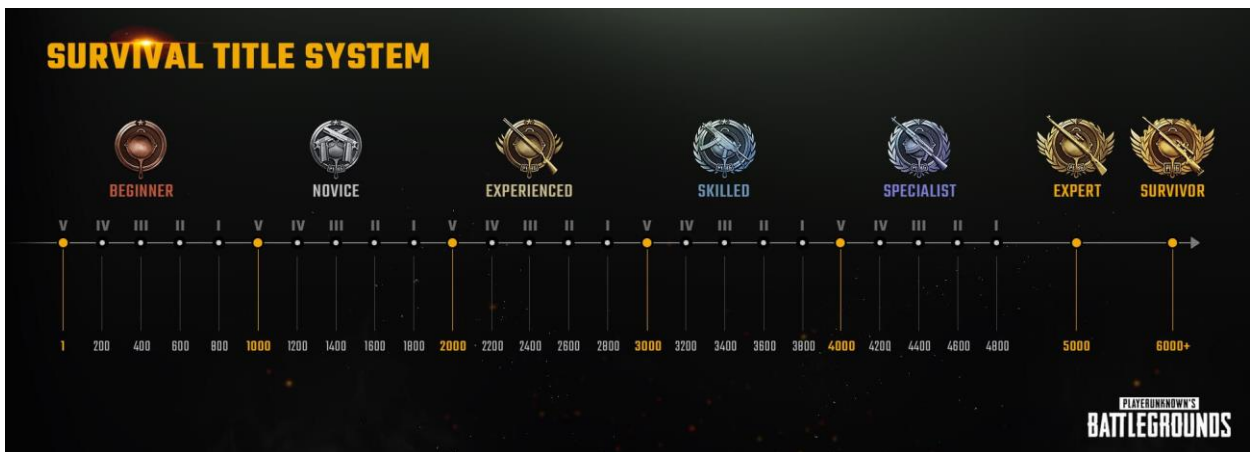


Рисунок 1.13 – Рівні рейтингової системи в грі PUBG

Також, в рейтинговому режимі, на відміну від звичайної битви, діють певні обмеження, які спрямовані на зрівняння всіх гравців. А саме:

1. Максимум 64 гравця в одній битві.
2. Тільки три певні карти, які доступні для гри. Обираються випадковим чином.
3. Відсутність так званих червоних зон.
4. Відсутність ботів.
5. Швидкість зміни зони збільшена, щоб у гравців було менше часу.

Це робиться для того, щоб всі були в рівних умовах і випадкові чинники не вплинули на результат гравців.

Оскільки в найпопулярніших онлайн іграх використовується рейтингова система підбору противника, то слід проаналізувати її, знайти всі її плюси та мінуси, а також, розробити свій власний алгоритм підбору, виправивши всі недоліки обраної системи підбору.

1.5 Поняття алгоритму

Алгоритм – це певний порядок скінченних команд, завдяки виконанням яких можна отримати бажаний результат [6]. Під командами потрібно

розуміти якісь конкретні дії, за допомогою яких отримується проміжний результат.

Слід розуміти також поняття виконавця алгоритму. Це може бути як жива істота, така як людина чи тварина, так і нежива, така як комп'ютер.

Виділяюся 6 основних властивостей алгоритмів:

1. Масовість. Тобто такий, який може виконувати багато задач в своєму роді.
2. Дискретність. Тобто розбиття на команди, які в свою чергу повинні бути чітко визначені.
3. Визначеність. Так звані команди алгоритму повинні бути чітко сформовані та не мати прихованого підтексту.
4. Результативність. Виконання кожної команди повинно дати якийсь результат.
5. Формальність. Виконавець алгоритму може навіть не розуміти змісту поставленої задачі, головне слідувати інструкції.
6. Скінченність. Тобто результат роботи алгоритму повинен бути отриманий за скінченну кількість команд [19].

Алгоритм можна описати декількома способами. А саме: схемою, словесно, за допомогою малюнку чи графіку. Розглянемо основний спосіб опису, у вигляді блок-схеми. На рисунку 1.14 можна побачити умовні позначення блоків для опису алгоритму.

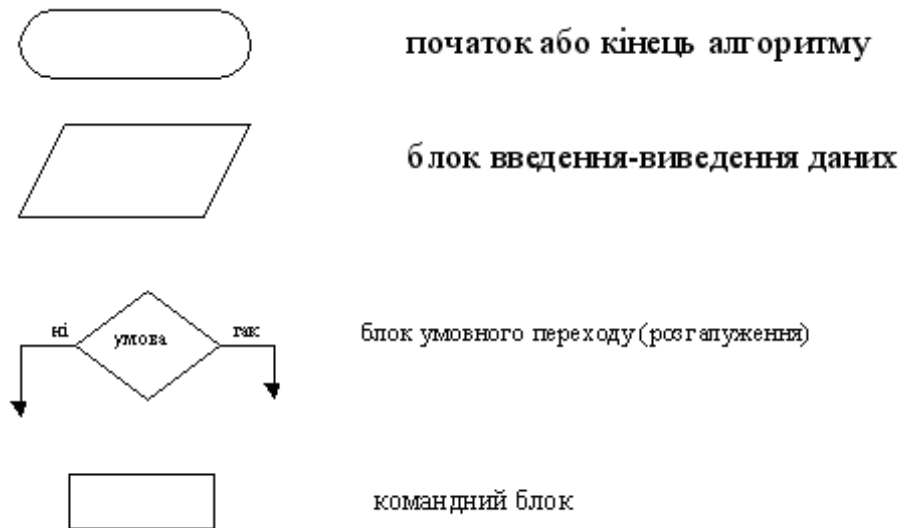


Рисунок 1.14 – Основні позначення алгоритмів на схемах

Використовуючи дані позначення, можна з легкістю описати якийсь примітивний алгоритм. Для прикладу розглянемо чищення картоплі. На рисунку 1.15 можна побачити описаний за допомогою блок-схеми алгоритм, за допомогою якого чистять картоплю. Алгоритм відповідає всім основним властивостям і є досить примітивний, тому що має маленьку кількість дій.

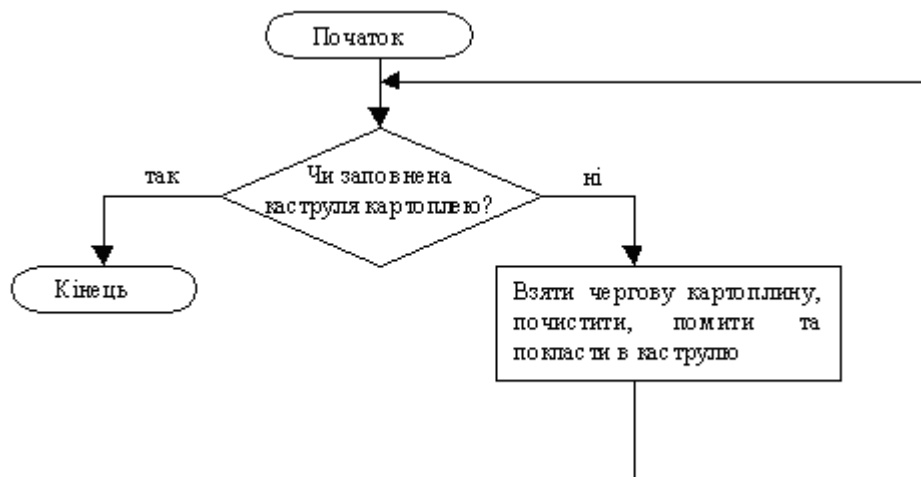


Рисунок 1.15 – Приклад алгоритму чищення картоплі

1.6 Алгоритми підбору

Існує безліч галузей, в яких застосовуються різноманітні алгоритми підбору. Від контекстної реклами, з якою споживачі зустрічаються кожного дня в своїх гаджетах, до прийому на роботу, де за певним алгоритмом відбирають кандидатів на вакансію.

У випадку з підбором, можна сказати, що це складний алгоритм, виконання якого потребує безліч часу. Розглянемо його як приклад. На рисунку 1.16 можна побачити приклад алгоритму підбору персоналу на підприємстві [7].

Якщо говорити про комп'ютерні або мобільні ігри, то можна сказати, що нема жодних, які б обійшлись без різного роду алгоритмів. Як приклад, алгоритм підбору противника. Мабуть в кожній онлайн грі є алгоритм, за допомогою якого гравцю підбирають противника. Це може бути як живий гравець, так і машина, штучний інтелект, бот, який створює спеціально для гравця неіснуючого противника, основується на параметрах самого гравця.

Кожна гра унікальна, тому виникає проблема з написанням загального алгоритму, який був би універсальним і підходив для всіх. Кожній компанії доводиться підганяти алгоритм під свої потреби.

Як приклад, візьмемо жанр MMORPG. В цьому випадку слід урахувувати безліч факторів. Такі як:

1. Характеристики персонажа.
2. Спорядження персонажа.
3. Статистику перемог та поразок.
4. Загальний рейтинг гравця.
5. Лігу гравця.

Таким чином, можна сказати, що знаючи основні особливості жанру, можна розробити універсальний алгоритм для конкретного жанру ігор.

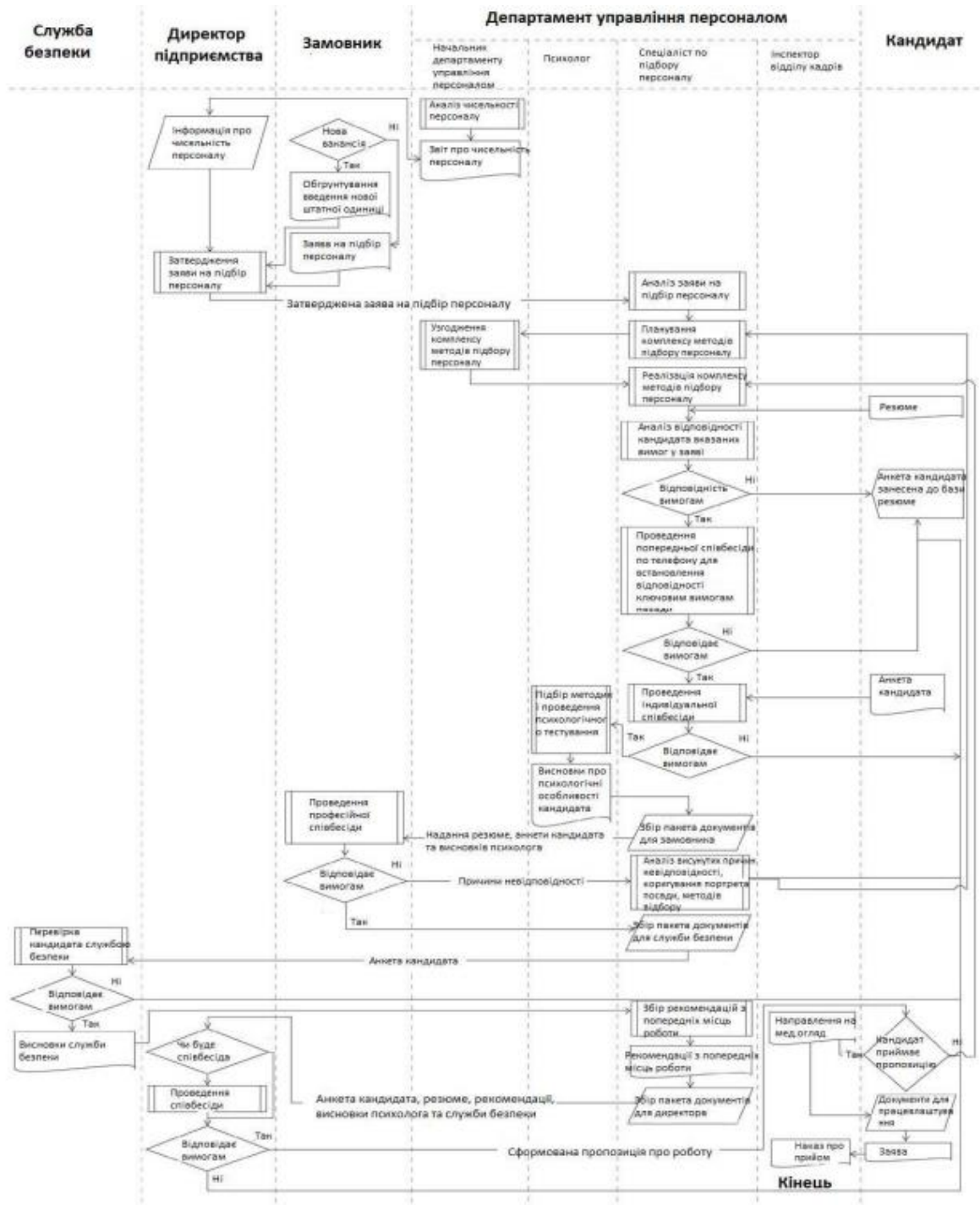


Рисунок 1.16 – Алгоритм підбору персоналу

1.7 Аналіз алгоритму підбору противника

1.7.1 Рейтингова система підбору гравця

Рейтинг підбору гравця (MMR – Matchmaking Rating) – це показник, за допомогою якого визначається особистий рейтинг гравця, його навички та вміння грати.

Його основні властивості:

1. Кількість отриманих балів за перемогу мінімум в два рази менша за кількість втрачених балів за поразку.
2. Чим більша загальна кількість балів – тим менша кількість отриманих балів за перемогу. І навпаки, чим менша загальна кількість балів – тим більша кількість отриманих балів за перемогу.
3. Захист від втрати балів при певних обставинах. Таких як вихід з гри через поганий інтернет, недобросовісних союзників, які посварились ще до початку гри або раптове вимкнення світла.
4. Накопичувальна система кількості перемог або поразок підряд. Тобто, чим більше гравець програє – тим слабших противників йому підбирає. І навпаки, чим більше він приграє – тим сильніших противників йому підбирає.

Всі проаналізовані ігри використовують саме рейтингову систему для підбору противника, адже вона є оптимальною. Але в усіх випадках вона відрізняється, тому що кожна гра особлива і тому потрібно його модифікувати під певні вимоги.

Рейтингова система підбору противника є стандартним рішенням для багатьох онлайн ігор, так як це дозволяє регулювати взаємну справедливість між гравцями та забезпечувати цілісність системи рейтингу. Однак, цей алгоритм має свої недоліки. Наприклад, ранній поразки можуть суттєво вплинути на подальший рейтинг гравця і викликати несправедливість у матчах. Також, цей алгоритм може не зберігати рівномірність рівнів

складності для різних гравців, що також може викликати незадоволення гравців.

Щоб виправити ці недоліки, можна розробити свій власний алгоритм підбору противника, який буде більш справедливим і більш задовільнять потреби гравців.

1.7.2 Математичний опис алгоритму

MMR є математичним моделювання рейтингу гравців, на основі якої кожен гравець має свій рейтинг, який відображає його силу. Рейтинг кожного гравця є числом, і може змінюватися від ігри до ігри в залежності від результату цієї ігри.

MMR використовує математичну функцію, названу функцією Ело, яка приймає в якості вхідних даних рейтинги гравців, що грають в даній грі, і повертає ймовірність перемоги кожного із гравців. Функція Ело може бути виражена так:

$$E(r1, r2) = 1 / (1 + 10^{((r2 - r1)/400)}) \quad (1.1),$$

де $r1$ – рейтинг першого гравця, $r2$ – рейтинг другого гравця. Ця функція повертає ймовірність перемоги першого гравця.

Після кожної гри використовується формула для оновлення рейтингу кожного гравця. Наприклад, якщо перший гравець переміг і має рейтинг $R1$, а другий гравець має рейтинг $R2$, то оновлення рейтингу буде виконуватися так:

$$R1' = R1 + K * (S - E(R1, R2)) \quad (1.2)$$

$$R2' = R2 + K * (L - E(R2, R1)) \quad (1.3),$$

де:

1. K – константа, яка контролює, наскільки сильно рейтинг гравця змінюється після кожної гри.
2. S – кількість балів, які гравець отримав за перемогу.
3. L – кількість балів, які гравець отримав за поразку.

Цей алгоритм використовує статистичні дані з кожної гри для постійного оновлення рейтингів гравців, щоб забезпечити максимально ефективне розподілення гравців з подібним рівнем навиків.

Алгоритм рейтингового підбору може бути налаштований за допомогою різних параметрів, таких як константа K , функція Ело та базовий рейтинг для нових гравців, які вперше приєднуються до системи.

Важливо зауважити, що цей алгоритм не є єдиним та є багато інших варіантів алгоритму рейтингового підбору, які можуть бути використовувані в залежності від вимог і особливостей ігрової платформи, метод рейтингового підбору може бути налаштований для більш ефективної роботи. Також можуть використовуватися інші функції рейтингу, як наприклад, функція головного моменту, котра може давати більш точні результати рейтингу для гравців з високим рівнем навиків.

Також можна використовувати комбінації різних методів для досягнення більш ефективного підбору рейтингу гравців, які можуть враховувати різні фактори, такі як статистика гри, географічне розташування, тощо.

Загалом, рейтинговий підбір – це комплексний процес, який використовує комбінацію математичних моделей, статистики і алгоритмів для обчислення рейтингу гравців та підбору одного гравця для гри з іншими гравцями з подібним рівнем навиків. Рейтинговий підбір дозволяє забезпечити більш справедливу і захоплюючу гру, і дозволяє кожному гравцеві зіграти з іншими гравцями з подібним рівнем навиків [16].

2 ПРОЕКТУВАННЯ ТА РОЗРОБКА АЛГОРИТМУ ПІДБОРУ ПРОТИВНИКА

2.1 Концепція та сценарій алгоритму

Концепція алгоритму полягає у тому, щоб забезпечити гравцям можливість позмагатися з однаковими за рівнем сили гравцями, які мають подібні характеристики та шанс на перемогу в бою.

Сценарій алгоритму може включати ряд таких кроків:

1. Інформація про гравців. Їх рівень, досвід, статистика, інвентар характеристики.
2. Групування гравців за схожими характеристиками.
3. Змагання гравців з однієї групи з гравцями з іншої групи з однаковим рівнем.
4. Збільшення бажання змагатися з іншими гравцями.

Це є лише один з можливих підходів, так як реалізація алгоритму може варіюватися в залежності від конкретної гри і її вимог. Наприклад, у деяких іграх можуть бути додаткові фактори, такі як командна гра або елементи PvP, які вимагають більш детальної обробки даних гравців. Алгоритм може також оновлюватися і адаптуватися для підвищення якості і ефективності змагань між гравцями.

Також, існує багато різних стратегій, які можуть використовуватися для змагання гравців у MMORPG. Одна з них – система «Quickmatch», яка використовує мінімальну кількість інформації про гравця, таку як його рівень та статистику, для швидкого бою з схожими за рівнем гравцями.

Інша стратегія – система «Matchmaking on demand», де гравець може самостійно вибрати опції для своєї гри такі як рівень складності, командна гра або одиночна гра та система буде здійснювати пошук противників відповідно до обраних опцій.

Ще одна стратегія – використання "ELO rating system", який поєднує дані про досвід і статистику гравця, щоб оцінити їхню силу позмагатися з гравцями з схожим рейтингом.

Це тільки декілька з багатьох стратегій, які можуть використовуватися для змагання гравців у MMORPG, і кожен з них може мати свої переваги і недоліки.

Важливо пам'ятати, що навіть самі найбільш оптимізовані алгоритми не можуть забезпечити чесну та справедливу гру для всіх гравців, часто гравці можуть очікувати від гри зовсім інше, мати особисті проблеми або банально погані навички і це все впливає на задоволення від гри.

Тому, розробники гри можуть використовувати комбінації різних стратегій та технологій для створення більш ефективного алгоритму підбору гравців. Наприклад, використання машинного навчання для аналізу даних гравців може дозволити більш точно визначати схожість гравців і покращувати якість підбору. Розробники також можуть використовувати форму зворотного зв'язку для поліпшення алгоритму підбору на основі реальних вимог гравців.

В загалом, концепція та сценарій алгоритму підбору в MMORPG полягає у забезпеченні гравцям можливість зіграти з схожим за рівнем та характеристиками гравцями. Розробники можуть використовувати різні стратегії та технології для максимізації якості і ефективності підбору, включаючи зменшення часу підбору, збільшення рівня конкуренції і задоволення гравців.

2.2 Підготовка елементів алгоритму

2.2.1 Модель персонажа та його сила

Одним з основних елементів математичної моделі є персонаж. Він може мати різні характеристики, такі як сила, спритність, інтелект та життя. Ці характеристики можуть бути змінені підвищенням рівня, квестами та

спорядження. Рівень може бути обчислений на основі досвіду, який отримує гравець у процесі гри, за допомогою формули:

$$EXP = BASE_EXP + LEVEL * EXP_MULTIPLAYER \quad (2.1)$$

Іншим важливим елементом може бути рюкзак гравця, який може містити значення здоров'я, мана, статистику атаки та захисту, а також інші елементи, які можуть впливати на гру. Цей рюкзак може бути змінений за допомогою квестів, купівлею/продажом предметів, тощо.

Також можуть бути режими бою та механіки, які використовують ці математичні моделі для опрацювання боїв між персонажами та монстрами. Наприклад, можна використовувати формули, які обчислюють відсоток попадання атаки і пошкодження, які може нанести персонаж, на основі характеристик і спорядження.

На рисунку 2.1 зображено графічну модель персонажа.



Nickname	♥ Welcome=)
People	Sheyvane
Health	♥ 3365...3365
Damage	⚡ 251...270
Accuracy	🔥 111...130
Armor	🛡️ 176
Agility	🏃 46...65
Stunning	🔪 108...127
Block	🛡️ 205
Bonuses	(614)
Secret trick 1	🔪 🔪
Secret trick 2	🔪 🔪 🔪
Secret trick 3	🔪 🔪 🔪 🔪
Reputation	Good
Reputation	67/0
Rank	Champion
Glory	31107/40000
Exp.	2473394 / 2863500(64%)
Endurance	618/618
Tournament rating	0

Рисунок 2.1 – Графічна модель персонажа

В залежності від конкретної гри, модель може містити більше або менше компонентів, але зазвичай вона складається з статистики персонажа, його рівня, характеристик та спорядження.

Також, можуть бути механіки взаємодії між гравцями, такі як групові завдання та групові бої. Ці механіки також можуть бути модельовані математично, наприклад, обчислення ефективності групи в залежності від рівня та статистики її членів, або математичне моделювання інтерактивності між гравцями у чаті гри.

Також, можуть бути різні системи економіки гри, які можуть включати в себе виробництво ресурсів, торгівлю, купівлю/продаж товарів і валюту. Ці системи також можуть бути модельовані математично, наприклад, за допомогою систем суперечок і економетрії для опрацювання взаємодії гравців та ресурсів у грі.

Оскільки MMORPG є складним та розвиненим типом відео-ігор, то математична модель може бути дуже складною та містити багато різноманітних компонентів. Це може включати в себе математичні моделі для опрацювання гри самої по собі, а також моделі для взаємодії гравців між собою та з системою гри. Також можуть бути розроблені математичні моделі для прогнозування поведінки гравців, такі як прогнозування рентабельності гри, або моделі для оптимізації геймплею.

Ще одною важливою математичною моделлю є моделі для обчислення ефективності команди гравців під час групового завдання. Ці моделі можуть використовуватися для оптимізації групових рівнів та боїв з босом так, щоб враховувати статистику та спорядження кожного члена команди і забезпечити ефективну роботу команди як в цілому. Ці моделі можуть також враховувати ролі кожного члена команди, такі як танк, лікар чи інші і обчислювати ефективність команди як в цілому, так і кожного її члена.

У загальному, математична модель гравця в MMORPG є комплексною та залежить від конкретної гри та її механік. Вона може включати в себе багато

різних компонентів, які можуть бути модельовані математично та використовуватися для опрацювання різних аспектів гри.

Розглянемо загальний приклад моделі персонажа в типових MMORPG іграх.

На рисунку 2.2 зображено текстову модель персонажа [8].



Рисунок 2.2 – Приклад текстової моделі персонажа в MMORPG іграх [8]

Це базова модель, яку можна застосувати до всіх ігор, адже тут описані обов'язкові параметри, які є у всіх іграх жанру MMORPG, тому вона буде використана при розробці алгоритму.

Алгоритм обчислення сили персонажа в багатокористувацькій онлайн грі жанру MMORPG може бути реалізований наступним чином:

1. Визначення базової шкоди персонажа на основі характеристики "шкода".
2. Обчислення додаткової шкоди на основі характеристики "точність" і рівня противника.
3. Визначення збереження шкоди на основі характеристики "броня" і рівня противника.
4. Обчислення шансу ухилитись від удару на основі характеристики "ухилення".
5. Обчислення шансу блоку на основі характеристики "блок".

Сумування всіх обчислених значень і виконання додаткових коефіцієнтів для отримання кінцевого результату сили персонажа

Реалізація алгоритму може відрізнитися в залежності від конкретної гри і системи боя. Наприклад, можуть бути включені додаткові характеристики, такі як мана, здоров'я, магія, зв'язки з іншими персонажами та інші.

Також, система може бути більш складною з багатьма додатковими параметрами які впливають на силу персонажа, наприклад збільшення пошкодження якщо противник оглушений або накладання різних ефектів від спорядження як на гравця, так і противника.

Цей алгоритм може бути постійно змінюваний і дороблятися розробниками для забезпечення балансу в MMORPG грі.

Формула для знаходження сили персонажа може бути різною в залежності від конкретної гри і системи балансу. Однак основними компонентами які можуть бути використані для розрахунку сили можуть бути:

$$F = damage + accuracy * enemy_level + armor * enemy_level + evasion + block \quad (2.2),$$

де:

1. F – сила персонажа.
2. Damage – базова шкода персонажа.
3. Accuracy – модифікатор влучності.
4. Enemy_level – рівень противника.
5. Armor – броня персонажа.
6. Evasion – шанс ухилитись.
7. Block – шанс блоку.

Також, можуть бути додані додаткові коефіцієнти для більш точного розрахунку сили, наприклад коефіцієнт здоров'я, мана або магії та інших.

Зауважимо, що це лише приклад, формула розрахунку сили у будь-якій грі може бути більш складною або спрощеною в залежності від конкретної гри. Також, може бути дороблені деякі параметри які впливають на силу персонажа або додані нові.

2.2.2 Модель рюкзака та його сила

Модель рюкзака в іграх – це система, яка керує персонажем з погляду спорядження. В іграх з великою кількістю гравців ця система може включати в себе вміст рюкзака, місця для обладнання, вибір спорядження, систему перетворення та обміну обладнання, тощо.

Рюкзак у MMORPG грі може мати різні розміри, кількість місця і обмеження на типи предметів, які можна зберігати в ньому. Рюкзак може бути доступним для всіх персонажів, або окремо для кожного з них.

В деяких іграх, рюкзак може бути розширеним за допомогою додаткового місця або розширення розміру, що дозволяє персонажу зберігати більше предметів.

Модель рюкзака також може включати систему спорядження, де персонаж може збирати і комбінувати різні предмети для створення нового обладнання або потенційно кращого. Також може бути реалізована система знищення обладнання, де обладнання може розпадатися або втрачати свою ефективність з часом або після використання.

Рюкзак також може мати систему категорій для показу предметів, які можуть бути відсортовані за типом, рівнем або іншими критеріями.

Окрім цього, рюкзак може мати систему доступу, де деякі місця можуть бути доступні тільки для персонажа або для команди, яка грає разом з ним.

В цілому модель рюкзака в MMORPG грі є важливим чинником для управління спорядженням, особливо коли гравці повинні пройти через різноманітні зони або боротьбу з різними монстрами, де потрібна добре спорядження і підготовка. Крім того, модель рюкзака також може впливати на механіку економіки ігри, яка може включати купівлю, продаж та обмін обладнання між гравцями.

Додатково, модель рюкзака може впливати на стратегію гравців і може використовуватись в якості метода обмеження і керування монополією ресурсів, які можуть бути використані для покращення спорядження.

В деяких іграх рюкзак є ключовим елементом для перемоги в бою і може бути складним інструментом, який треба завжди покращувати для досягання кращих результатів.

Загалом, модель рюкзака в MMORPG іграх є складним і важливим елементом, який може впливати на гру, гравців та їхні стратегії і виконання завдання.

На рисунку 2.3 зображено загальний вигляд рюкзака в MMORPG іграх.



Рисунок 2.3 – загальний вигляд рюкзака в MMORPG іграх.

Опишемо загальну формулу, яка буде підходити для більшості ігор. Для цього будемо знаходити такий параметр як сила. Його в майбутньому буде застосовано в формулі для вирахування індивідуального рейтингу персонажа, завдяки якому можна буде знайти вже противника для нього.

Сила гравця може бути змінюватися на основі різних характеристик, таких як рівень, статистика, спорядження, так само як і на основі інших факторів. Формула для знаходження сили гравця може бути різною для різних ігор.

Однак, якщо мова йде про формулу знаходження сили гравця аналізуючи його рюкзак, можна використовувати таку формулу:

$$S = (vhead + vhands + vbody + vlegs + vshoes + vweapon + vshield + vamulet + vring1 + vring2 + vpotion) * (1 + bhead + bhands + bbody + blegs + bshoes + bweapon + bshield + bamulet + bring1 + bring2) \quad (2.3),$$

де S – сила гравця, v – значення спорядження в рюкзаку, b – бонус спорядження. Значення бонусу може бути визначене за допомогою вхідних параметрів, які вказують адміністратори гри або інших методів.

Це лише один з прикладів, як можна знайти силу гравця, як завжди, можливі різні варіанти і моделі для розрахунку.

2.2.3 Коефіцієнт перемог та поразок

Коефіцієнт перемоги та поразок персонажа може бути обчислений за допомогою наступного алгоритму:

1. Збереження використання гравцем кожної позиції та результату бою в базі даних.
2. Розрахунок коефіцієнта перемоги для кожної позиції, який є відношенням кількості перемог до загальної кількості боїв для цієї позиції.
3. Розрахунок коефіцієнта поразок для кожного гравця, який є відношенням кількості поразок до загальної кількості боїв для цього гравця.
4. Виведення результатів на екран для кожного гравця, які містять коефіцієнт перемоги та коефіцієнт поразок для кожної позиції, яку вони використовували.

Важливо зауважити, що цей алгоритм може бути змінений та модифікований в залежності від потреб та вимог гри. Наприклад, розрахунок коефіцієнту може бути об'єднаний з іншими показниками, такими як рівень, досвід, або звання персонажа. Також можуть бути використані додаткові

критерії для оцінки ефективності гравця, наприклад показники навчання та контроль здоров'я персонажа.

Також можуть бути використані розподілення потенційних противників по рівнях та групам, щоб обчислити коефіцієнт перемоги ще більш точно. Також може бути використана статистика групової гри, щоб оцінити ефективність командної гри та обчислити коефіцієнт перемоги для кожного гравця в команді.

Результати можуть бути відображені в загальному рейтингу, щоб гравці могли оцінити свої власні навички та порівняти свою ефективність з іншими гравцями. Алгоритм може бути постійно модифікований і оновлений, щоб забезпечити якість та точність результатів.

Коефіцієнт перемоги може бути обчислений за допомогою формули:

$$K = (W / T) * 100\% \quad (2.4),$$

де:

1. K – Коефіцієнт перемоги.
2. W – кількість переможних боїв для даної позиції.
3. T – загальна кількість боїв для даної позиції.

Коефіцієнт поразок може бути обчислений за допомогою формули:

$$L = (L / T) * 100\% \quad (2.5),$$

де:

1. L – коефіцієнт поразок.
2. L – кількість поразок для даного гравця.
3. T – загальна кількість боїв для даного гравця.

В залежності від гри та її системи розрахунків можуть бути використані різні формули для розрахунку коефіцієнтів, або додані додаткові фактори для оцінки результату.

Також можуть бути додані додаткові фактори для оцінки результату, наприклад обстановка бою, об'єктивні цілі, склад команди, тощо. Це може потребувати модифікації формули для розрахунку коефіцієнту перемоги та поразок. Наприклад, може бути використана формула яка враховує кількість убитих ворогів, та інші показники бою.

Також важливо враховувати, що коефіцієнти перемоги та поразок можуть бути використані лише як індикатор ефективності гравця, але не як єдиний критерій для оцінки результату. Вони повинні використовуватися разом з іншими показниками, такі як рівень, досвід, тактика, та спільна гра в команді.

2.3 Внутрішня ігрова логіка

В MMORPG іграх гравці взаємодіють один з одним у віртуальному світі, який симулює реальне середовище. Внутрішня логіка гри, або механіка гри, регулює те, як гравці взаємодіють з цим віртуальним світом та один з одним.

Одним з важливих аспектів внутрішньої логіки гри є правила створення та розвитку персонажа. Гравці зазвичай починають з створення персонажа, який є цифровим аватаром, який представляє їх в грі. Гравці можуть налаштовувати зовнішній вигляд, здібності та статистику свого персонажа. Коли гравці грають в гру, їх персонажі отримують бали досвіду, які можна використовувати для підняття рівня та отримання нових здібностей або предметів.

Інший важливий аспект – це економіка гри та те, як дії гравців впливають та взаємодіють з нею. Це може включати в себе те, як гравці можуть купувати, продавати та торгувати предметами в грі, як змінюється вартість різних предметів та як дії гравців впливають на економіку.

Також система бою є важливою. Гравці можуть боротися проти NPC або проти інших гравців у PVP. Правила, які регулюють бій, такі як сила, точність, спритність, броня, блок та здібності, визначаються внутрішньою логікою гри.

Додатково, MMORPG іграх часто є система квестів, з завданнями, які гравці можуть виконувати, щоб отримати бали досвіду та інші нагороди. Внутрішня логіка контролює те, як гравці можуть взаємодіяти з NPC, приймати та виконувати квести та переміщуватись по світу гри.

Нарешті, багато MMORPG мають соціальні системи, які дозволяють гравцям формувати групи та спільноти, обмінюватися повідомленнями один з одним та виконувати групові завдання разом. Внутрішня логіка контролює те, як такі соціальні взаємовідносини відбуваються.

Внутрішня логіка також включає рейтингову систему, яка оцінює гравців на основі їх досягнень та показників в грі. Це може включати в себе завершення спеціальних квестів, досягнення високого рівня, або перемоги в бою проти інших гравців. Рейтингова система допоможе гравцям порівнювати свої досягнення з іншими гравцями та мотивує їх до досягнення нових висот.

Ще одним важливим елементом внутрішньої логіки є система досягнень, це список завдання та задач, які гравці можуть виконувати, щоб отримати додаткові нагороди. Ці досягнення можуть бути пов'язані з різними аспектами гри, наприклад, спроба перемогти певного боса, досягти високого рівня, або отримання високих результатів в PVP бою. Досягнення додають додатковий рівень глибини та мотивації для гравців, щоб виконувати нові завдання та досягати нові цілі.

Також, може бути реалізована система прокачки, що дозволяє гравцям навчати своїх персонажів та піднімати їх статистику за допомогою балів досвіду, які вони отримують. Це дає гравцям більше контролю над розвитком своїх персонажів та дозволяє їм налаштовувати їх для певного стилю гри.

Це може бути карта міста, перебування в локаціях подібних до печер чи джунглів, або навіть космічні кораблі. Ці локації містять різні завдання, монстрів і предмети, які гравці можуть здобути, щоб ставати сильнішими. В

більшості внутрішніх ігрових локацій присутні інші гравці, тому вони також служать місцем для соціалізації і співпраці гравців.

Внутрішні ігрові локації часто містять різні рівні складності, призначені для різних рівнів гравців. Наприклад, початківці можуть почати свою подорож в локаціях з низькою складністю, де вони можуть навчитися основам гри і повертатися до них пізніше для збору ресурсів та монстрів. Більш досвідчені гравці можуть перейти до більш складних локацій, де вони можуть змагатися з більш сильними противниками і отримувати більш вигідні нагороди.

Внутрішні ігрові локації також можуть бути пов'язані з історією гри, і гравці можуть дізнаватися більше про її світ і персонажів, подорожуючи по різних локаціях. Вони також можуть виконувати завдання і квести, які пов'язані з історією і локацією, і допомагають вирішувати загадки і вирішувати проблеми в світі гри.

В цілому, внутрішня логіка MMORPG складається з багатьох різних елементів, які взаємодіють разом, щоб створити реалістичну та цікаву гру. Розвиток та створення персонажа, економіка, бій, квести, соціальні взаємовідносини, рейтингова система, система досягнень, система прокачки – це лише декілька з елементів, які допомагають створювати реалістичну та цікаву гру для гравців. Ці елементи можуть розроблятися розробниками гри та можуть мінятися в залежності від потреб та вимог гравців. Це дозволяє гравцям продовжувати відкривати нові можливості та досягати нові цілі в грі, захоплюючись нею.

3 АПРОБАЦІЯ РЕЗУЛЬТАТІВ РОБОТИ

3.1 Реалізація створеного алгоритму

3.1.1 Опис гравців та режимів гри

Для реалізації створеного алгоритму та проведення тестувань була обрана гра в жанрі MMORPG. По-перше, потрібно сказати, що в грі є чотири основні типи гравців.

Перший тип – це сам персонаж, який створений безпосередньо людиною. В нього є такі характеристики як: здоров'я, шкода, точність, броня, блок та ухиляння. Також, він носить на собі якісь речі, які посилюють його характеристики. Речі можна поділити на такі групи: Шлем, нагрудник, нарукавники, штани, взуття, зброя, щит, амулет, кільця та зілля. Всі ці речі знаходяться на ньому або в його рюкзаку. За потреби можна перевдягнутися у інші речі, змінивши цією дією характеристики персонажа. Останнім пунктом, що впливає на підбір, є статистика, яку містить кожен персонаж. Це кількість переможених ворогів, кількість смертей, кількість переможених монстрів та багато прихованих даних, таких як тенденція перемог та поразок, які впливають на підбір противника для персонажа.

На рисунку 3.1 зображений персонаж з гри.



Рисунок 3.1 – Персонаж з гри

На рисунку 3.2 зображені характеристики персонажа з урахуванням всіх одітих предметів на ньому.



Рисунок 3.2 – Характеристики персонажа

Другий тип – це боти, тобто штучний інтелект, який імітує справжніх гравців. Вони були створені за для збільшення загальної кількості гравців на сервері, збільшенні кількості персонажів у бою та збільшення важкості самого ігрового процесу. Вони абсолютно ніяк не відрізняються від справжніх гравців, також мають свої характеристики та предмети, які носять в своєму рюкзаку. Також вони мають такий самий вигляд, відрізнити їх можна хіба що поведінкою в бою.

Третій тип – це NPC, тобто не живі, мирні персонажі, які ніяк не можуть допомагати героям в бою чи змагатися з ними. Їх використовують для взаємодії з гравцем. Це може бути торгівля з гравцем, тобто покупка або продаж речей, які знаходяться у NPC та персонажа, проходження завдання, які NPC видає герою, за які герой отримує винагороду. Це може бути як ігрова валюта, так і предмети. Також це може бути карта скарбів або пропуск в локацію, де знаходяться різної складності та рівня монстри. На рисунку 3.3 зображений приклад отримання завдання від NPC.



Рисунок 3.3 – Отримання завдання від NPC

Четвертий тип – це монстри. Їх буває багато видів. Це можуть бути як якісь справжні істоти, типу собак, вовків та інших, так і несправжні істоти, таких як гноми, ельфи, дракони, тощо. В них вже нема рюкзака, як у персонажів, але є всі ті ж самі характеристики.

Алгоритм підбору гравця буде застосований до всіх режимів бою, адже важливо щоб кожен гравець отримував позитивний досвід гри. Основними режимами бою є:

1. Битва 1 на 1. Це рівна битва без можливості допомогти іншим гравцям.
2. Відкритий бій. Це битва, яка розпочинається як битва в режимі 1 на 1, але з можливістю допомоги.
3. Вживання. Бій, в якому кожен сам за себе. Виграє той, хто переможе всіх своїх суперників.
4. Битва з монстром. Бій, проти монстра. Зазвичай використовується для проходження різних завдань, які видають NPC.

На рисунку 3.4 зображений приклад битви з монстром.

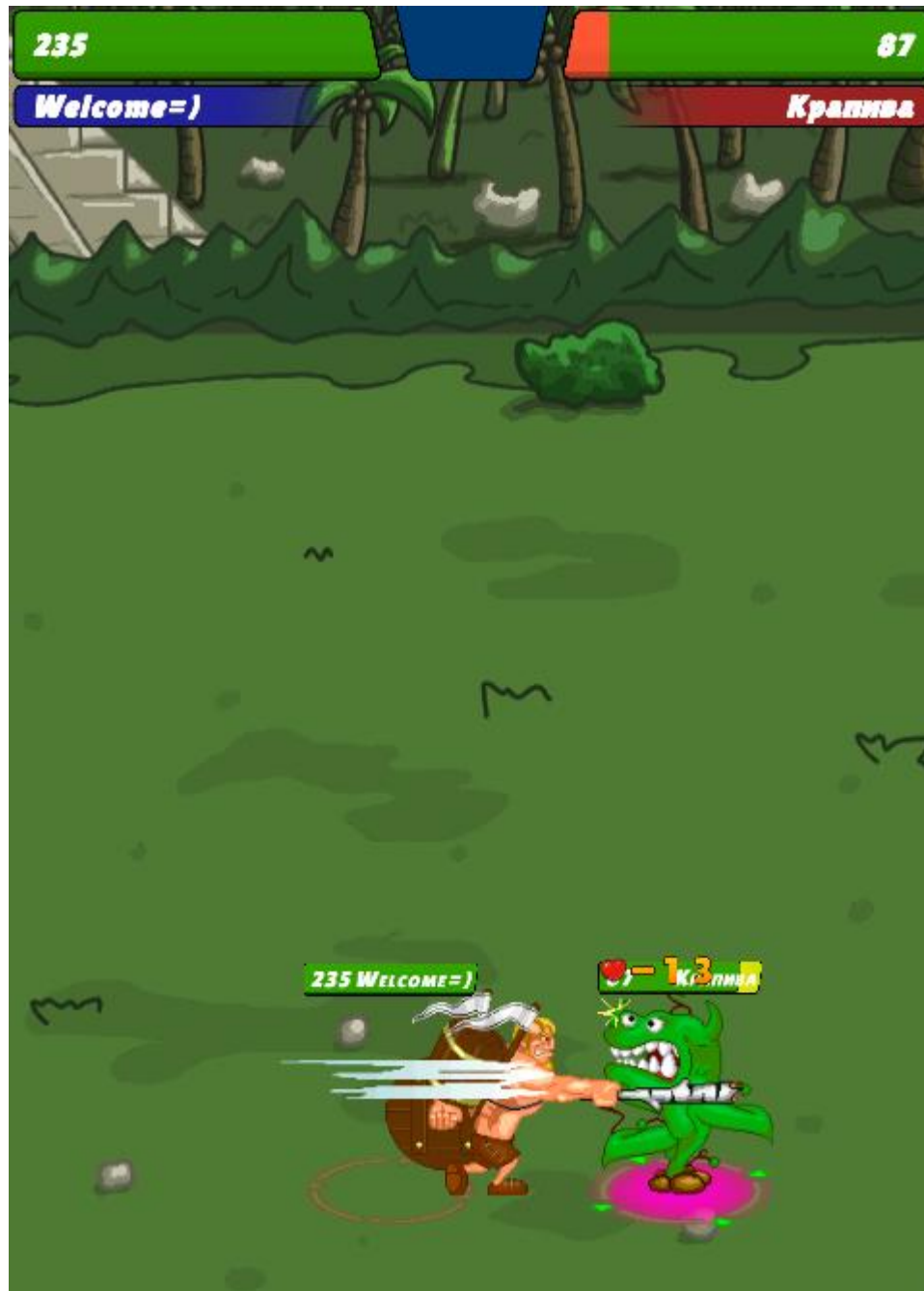


Рисунок 3.4 – Приклад битви з монстром

3.1.2 Характеристики підбору противника

Підбір противника здійснюється за допомогою розробленого алгоритму, який опирається на основні характеристики підбору. Розглянемо їх:

1. Рейтинг гравця. В грі реалізована система рейтингу гравців. Допустимий діапазон значень від 0 до 5000 балів. Якщо багато гравців досягають максимального значення, то його піднімають на

1000 балів. Всі нові гравці повинні пройти тестування на свій рівень майстерності гри. Воно складається з проходження різного виду завдань на витривалість та час, після чого гравцю видають його особистий рейтинг. Далі, гравцю доступні основні режими битви, в яких використовується алгоритм підбору противника. В залежності від отриманих результатів у бою, його рейтинг змінюється в більшу або меншу сторону.

2. Сила персонажа. В даному випадку використовуються дві основні формули, завдяки яким знаходиться кількість балів сили персонажа. По-перше, це сила самого персонажа, без урахуванням його предметів та рюкзака. По-друге, це сила рюкзака персонажа. Туди входять предмети, в які персонаж може переодягнутись між битвами, завдяки чому виграти проти певного типу монстрів чи інших гравців. По-третє, це сила одягнутих на героя предметів. Сюди входить зброя, щит, шлем, нарукавники, обладунок, взуття, кільця, намисто, зілля. Всі ці пункти застосовуються в формулі та впливають на кінцевий результат. На формулу впливає велика кількість чинників, які ж до всього ще й постійно оновлюються, додаються нові предмети в гру, через що значення може бути від 100 до нескінченної кількості балів.
3. Коефіцієнт перемог та поразок. Кожному персонажу при створенні присвоюється значення 1, після кожної битви він піднімається на 0.1 пункт, якщо герой виграє, або опускається на ті ж самі 0.1, якщо гравець програє битву. Завдяки цьому, гравців, які часто програють через своє невміння грати, будуть зустрічатись з такими ж самими гравцями, які програють. Саме тому їх шанси будуть рівними. Діапазон значень може бути від 0.1, коли гравець постійно програє, до нескінченної кількості балів, коли гравець постійно виграє битви.

Загальна формула підбору виглядає так:

$$P = \frac{R + P_{\Pi} + S}{3} \quad (3.1)$$

Якщо брати рейтинг гравця, то в гарному випадку вважається, що максимальна різниця не повинна перевищувати 200 балів, адже саме в такому випадку не порушується правило правильного підбору. Якщо ж відхилення більше, наприклад, 500, то це вважається поганим підбором, адже в такому випадку сили не рівні.

Для сили персонажа, в гарному випадку, вважається, що максимальна різниця не повинна перевищувати 500 балів, адже потрібно враховувати, що в грі є різні класи героя, тому сила персонажа може відрізнитися, що зумовлено особливостями класової системи гри. Якщо ж відхилення більше, то це вважається поганим підбором.

Для коефіцієнта перемог та поразок, в гарному випадку, максимальним допустимим відхиленням є 0.5 балів, адже він достатньо сильно впливає на формулу підбору противника. В поганому ж випадку, це може бути більше відхилення, яке суттєво вплине на кінцевий показник і підбір рівного противника може закінчитись невдало.

Потрібно сказати, що перш за все пошук противника йде за мінімальним відхиленням, щоб як найкраще підібрати суперника.

3.1.3 Коефіцієнт підбору гравця

Під коефіцієнтом підбору гравця слід розуміти модуль різниці між загальною кількістю балів P у формулі 3.1 між двома гравцями. Математично це можна описати так:

$$K = |P_1 - P_2| \quad (3.2),$$

де:

1. K – Шуканий коефіцієнт підбору.

2. P1 – Загальна кількість балів у першого персонажа.
3. P2 – Загальна кількість балів у другого персонажа.

Завдяки даному значенню, можна визначити на скільки сильно відрізняються два героя и зрозуміти, можна їх додавати в один бій чи ні. Використаємо табличку 3.1, в які зображено допустиме значення коефіцієнту підбору гравця. Дана таблиця отримана шляхом аналізу бої з різними значеннями коефіцієнту.

Таблиця 3.1 – Допустиме значення коефіцієнту підбору гравця

Коефіцієнти підбору персонажів	
301+	Низький
151-300	Середній
0-150	Високий

Як виявилось, допустиме значення відхилення дорівнює 150 балів. Тобто, якщо коефіцієнт підбору не перевищує його, то це вважається високою ступеню підбору, що свідчить про максимальну подібність персонажів у бою, тобто гарний підбір.

Якщо ж значення в межах від 151 до 300, то це вважається середнім ступенем підбору. Такий підбір не можна назвати вдалим, але завдяки випадковим попаданням по противнику, його все ж таки можна перемогти. Все

залежить від удачі. За такий бій дають більшу кількість балів рейтингу, адже шанс на перемогу менший.

Якщо ж коефіцієнт підбору дорівнює 301 та більше, то це можна вважати низьким ступенем підбору, тобто противник не рівний по силі, за рахунок чого противнику при перемозі дадуть менше балів рейтингу та ресурсів. В даному випадку, це поганий підбір противника.

3.2 Результати застосування алгоритму

Щоб довести ефективність роботи алгоритму, проведемо тестування підбору противника до застосування алгоритму, та, безпосередньо, після нього. Для цього виконаємо чотири тестові підбори. В таблиці 3.2 зображені дані першого тестового підбору.

Таблиця 3.2 – Перший тестовий підбір.

Підбір до застосування алгоритму			Підбір після застосування алгоритму		
	Гравець 1	Гравець 2	Підбір 1	Гравець 1	Гравець 2
Рейтинг	2000	2500		2000	2050
Загальна сила персонажа	500	800		500	600
Коефіцієнт перемог та поразок	0,8	1,1		0,8	0,9
Коефіцієнт підбору		400,15	Коефіцієнт підбору		75,05

Отже, до застосування алгоритму, коефіцієнт підбору знаходиться на відмітці 400,15. Це свідчить про поганий та неефективний підбір противника. Якщо ж дивитись на частину після застосування алгоритму, то можна відзначити, що всі показники не перевищують допустиму норму. Різниця

рейтингу всього 50 балів, що є нормою. Різниця в загальній силі знаходиться на відмітці 100 одиниць та також не перевищує допустиму норму. Коефіцієнт перемог та поразок знаходиться в межах норми. Розглянемо, як це виглядає, безпосередньо, у самій грі. На рисунку 3.5 зображені характеристики персонажів під час бою між собою після застосування алгоритму.

Герой 1 [18]				
				
2618	2618	225	155	
				
276	195	150	227	
▶ Переглянути вміст рюкзака				
Коефіцієнт підбору: 1951 бал				

Герой 2 [18]				
				
2512	2512	239	188	
				
290	173	165	210	
▶ Переглянути вміст рюкзака				
Коефіцієнт підбору: 2099 балів				

Рисунок 3.5 – Характеристики персонажів під час бою

Таким самим чином, проведемо ще 3 тестові підбори противника. В таблицях 3.3, 3.4 та 3.5 зображені наступні три тести.

Таблиця 3.3 – Тестування підбору противника

Підбір до застосування алгоритму			Підбір після застосування алгоритму			
	Гравець 1	Гравець 2	Підбір 2	Гравець 1	Гравець 2	
Рейтинг	3000	2000			3000	2900
Загальна сила персонажа	900	500			900	910
Коефіцієнт перемог та поразок	1,5	1,1			1,5	1,6
Коефіцієнт підбору		700,2	Коефіцієнт підбору		44,95	

У випадку з другим підбором, можна зробити такі ж самі висновки, як і з першим. Результат знаходиться в межах високої ефективності підбору.

Таблиця 3.4 – Тестування підбору противника

Підбір до застосування алгоритму			Підбір після застосування алгоритму			
	Гравець 1	Гравець 2	Підбір 3	Гравець 1	Гравець 2	
Рейтинг	1990	2440			1990	2090
Загальна сила персонажа	440	600			440	780
Коефіцієнт перемог та поразок	0,8	1,1			0,8	1,2
Коефіцієнт підбору		305,15	Коефіцієнт підбору		220,2	

Слід відмітити, що у випадку з третім підбором, коефіцієнт підбору противника знаходиться в середній степені підбору. Це свідчить про те, що підбір виявився не ідеальним, але все ж таки набагато кращим підбір без використання алгоритму.

Таблиця 3.5 – Тестування підбору противника

Підбір до застосування алгоритму			Підбір після застосування алгоритму		
	Гравець 1	Гравець 2	Підбір 4	Гравець 1	Гравець 2
Рейтинг	4100	3200		4100	4050
Загальна сила персонажа	1010	650		1010	850
Коефіцієнт перемог та поразок	1	1,4		1	1,2
Коефіцієнт підбору		629,8	Коефіцієнт підбору		104,9

У випадку з останнім тестом, підбір виявився вдалим, тому що коефіцієнт підбору знаходиться у високій степені підбору. Відповідно як тести 1 та 2.

Порівняємо отримані значення. Для цього зведемо їх в таблицю. В таблиці 3.6 порівнюються значення коефіцієнтів підбору в зроблених тестах.

Таблиця 3.6 – Порівняння коефіцієнтів підбору.

№ підбору	1	2	3	4
Коефіцієнти підбору до застосування алгоритму	400,15	700,2	305,15	629,8
Коефіцієнти підбору після застосування алгоритму	75,05	44,95	220,2	104,9

Також, можна зобразити все це у вигляді діаграми, щоб було більш наглядно на скільки покращується коефіцієнт підбору завдяки застосування алгоритму підбору противника.

На рисунку 3.6 зображено діаграму-порівняння коефіцієнтів підбору до та після застосування алгоритму.

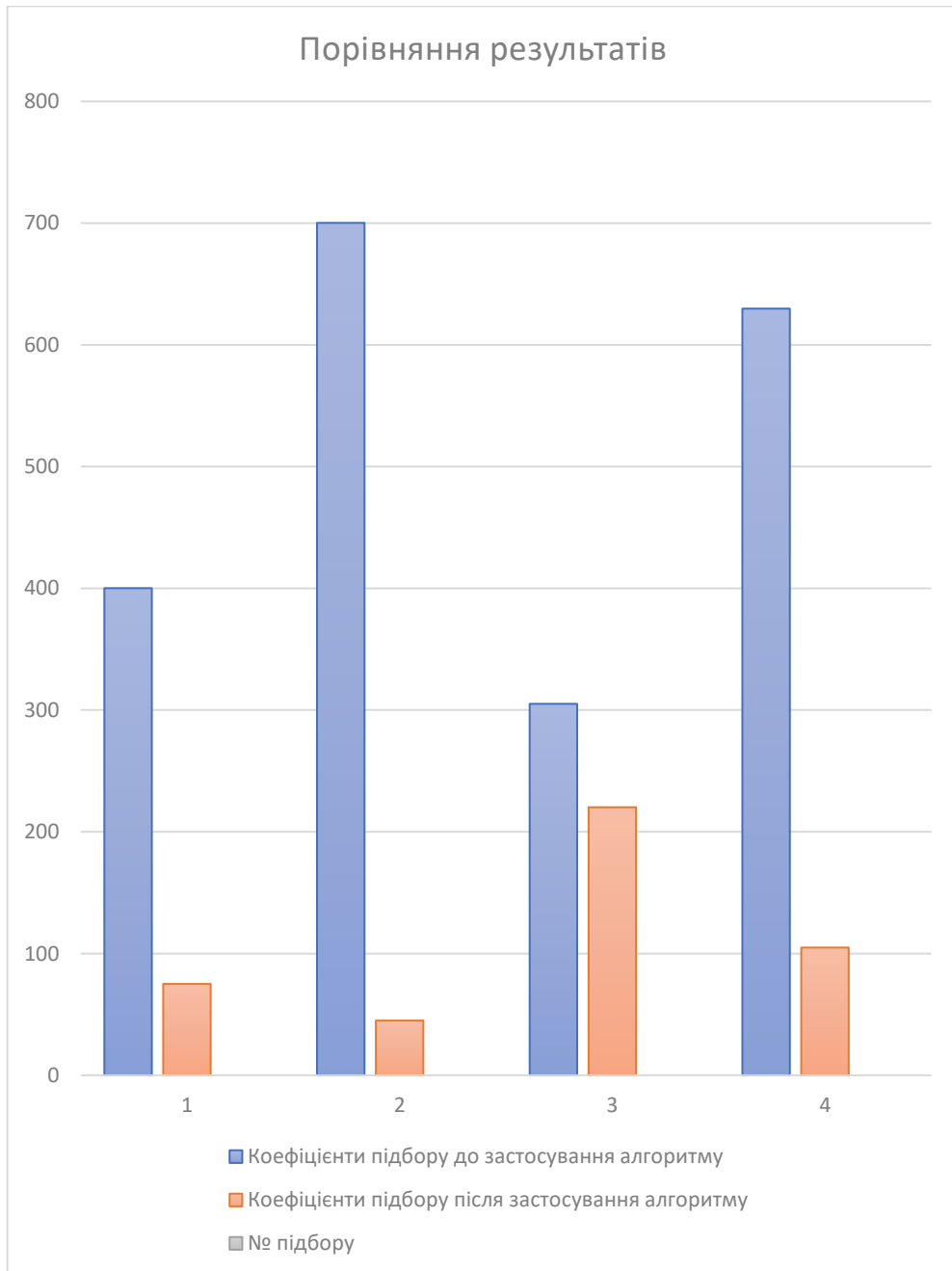


Рисунок 3.6 – Діаграма-порівняння коефіцієнтів підбору

Далі, вирахуємо похибку в підборі. Це допоможе нам знайти відсоток, на який покращилась точність підбору. Для цього потрібно використати формулу:

$$T = \frac{K_{\text{пі}} * 100\%}{K_{\text{до}}} \quad (3.3),$$

де:

1. T – Похибка в підборі.
2. $K_{пі}$ – Коефіцієнт підбору після застосування алгоритму.
3. $K_{до}$ – Коефіцієнт підбору до застосування алгоритму.

Знаючи похибку, можна обчислити точність підбору в %. Для цього слід використати формулу:

$$D = 100\% - T \quad (3.4),$$

де:

1. D - % точності підбору.
2. T – Похибка в підборі.

Далі, знайдемо середнє арифметичне точності підбору між всіма 4 проведеними тестуваннями. Для цього слід використати формулу середнього арифметичного.

$$X = \frac{D_1 + D_2 + D_3 + \dots + D_n}{n} \quad (3.5),$$

де:

1. X – Середнє арифметичне.
2. D -Точність підбору в %.
3. N – Кількість елементів.

Таким чином, можна отримати таблицю 3.7, в якій будуть представлені всі описані вище значенням із використанням указаних формул.

З цього випливає, що для першого тестування покращення склало 81%, для другого – 93%, для третього – 27% і для четвертого – 83%. Середнє значення, на яке покращився підбір – 71%.

Таблиця 3.7 – Таблиця з похибкою та точністю підбору.

	1	2	3	4
Похибка із застосуванням алгоритму, %	18,75547	6,4195944	72,1612	16,656081
Точність підбору персонажів, %	81,24453	93,5804056	27,8388	83,343919
Середнє значення усіх підборів, %	71,50190636			

Отже, можна зробити висновок, що алгоритм підбору противника для мультимедійної онлайн гри в жанрі MMORPG справляється з поставленою задачею, покращуючи точність підбору противника в середньому на 71%, про що свідчать дані дослідження.

ВИСНОВКИ

В результаті виконання магістерської роботи було розроблено алгоритм підбору противника в мультимедійній онлайн грі в жанрі MMORPG.

При виконанні роботи були вирішені наступні задачі:

1. Досліджено ринок комп'ютерних ігор, його основні тенденції як в Україні, так і світі в цілому. Оглянуті найпопулярніші онлайн ігри сьогодення для дослідження їх алгоритмів.
2. Розглянуті моделі персонажа та рюкзака. Їх основні властивості та характеристики. Досліджений та розроблений алгоритм для знаходження сили використовуючи дані моделі. Також, знайдено коефіцієнт перемог та поразок, завдяки якому можна зрозуміти тенденцію успіхів кожного із гравців.
3. Досліджено алгоритми, їхню загальну характеристику, наведений приклад алгоритму підбору.
4. Спроектований та розроблений алгоритм підбору противника. Описаний його принцип дії.
5. Проведено апробацію розробленого алгоритму. Доведено його ефективність.

Таким чином, результатом магістерської роботи є повністю виконаний поставлений план задач.

ПЕРЕЛІК ПОСИЛАНЬ

1. Apperley T. H. Genre and game studies: Toward a critical approach to video game genres / T. H. Apperley // Simulation & Gaming. – 2006. – Vol. 37. – No. 1. – P. 11 – 56.
2. Clearwater D. What Defines Videogame Genre? Thinking about Genre Study after the Great Divide / D. Clearwater // The Journal of the Canadian Game Studies Association. – 2011. – No. 5. – s. 45 – 87.
3. Counter-Strike: Global Offensive. URL: <https://blog.counter-strike.net/>. (Дата звернення: 24.11.2022р.).
4. Dota 2. URL: <https://www.dota2.com/home>. (Дата звернення: 23.11.2022р.).
5. Gregory Jason. Game Engine Architecture. – New York, CRC Press. – 2009. – No. 5. – s. 9 – 44.
6. McShaffry, Mike, Graham David. Game coding complete: Fourth Edition. – Boston, Course Technology. – 2013. – 112s.
7. MindMeister. URL: <https://www.mindmeister.com/>. (Дата звернення: 10.12.2022р.).
8. Nystrom R., Game Programming Patterns – Kindle Edition, 2014р. –115ст.
9. Perry D., David Perry on Game Design: A Brainstorming Toolbox Rusel – DeMaria , 2008 р. – 222 ст.
10. PUBG: BATTLEGROUNDS. URL: https://store.steampowered.com/app/578080/PUBG_BATTLEGROUNDS/?l=ukrainian. (Дата звернення: 24.11.2022р.).
11. Where does gaming go next? URL: <https://games.withgoogle.com/reports/beyondreport/>. (Дата звернення: 20.11.2022р.).
12. Wong J., Metaballs and Marching Squares. URL: <http://jamiewong.com/2014/08/19/metaballs-and-marching-squares/>. (Дата звернення: 12.12.2022р.).

13. Алгоритм. Поняття алгоритму. Його особливості та форми. URL: <https://sites.google.com/site/vchimoinformatikurazom/zavdanna/ponatta-algoritmu>. (Дата звернення: 28.11.2022р.).
14. Кушнерьов О.С, Каца М.О. Роль ігрової індустрії як складової інформаційних технологій. 2021.
15. Основні властивості алгоритму. URL: <https://step.org.ua/konspekt/algoritm/tema1>. (Дата звернення: 01.12.2022р.).
16. Под'ячев А. Д. Інтелектуальна система моделювання поведінки персонажів в комп'ютерних іграх. 2022.
17. Романюк О. Аналіз ринку комп'ютерних ігор. 2021.
18. Управління процесом рекрутингу на підприємствах. URL: http://www.economy.nayka.com.ua/pdf/1_2018/47.pdf. (Дата звернення: 07.12.2022р.).
19. Шевченко, Ю. О. *Моделювання проблем класифікації з підбором оптимального алгоритму*. Суми.
20. Шолудько О. Аспекти економічного розвитку України у сфері ігрової індустрії в умовах глобалізації. 2022, с.58.

Додаток А

Основний набір функціоналу розробленого алгоритму

```

<?php

require_once 'bd.php';
require_once      $_SERVER['DOCUMENT_ROOT']
'/functions/genbotpar.php';
ignore_user_abort(true);

register_shutdown_function('shutdown_d');
if      (apcu_exists($_SERVER['DOCUMENT_ROOT']
'block_odnopotok_duel_1_1_2')) {
    exit(0);
} else {
    apcu_store($_SERVER['DOCUMENT_ROOT']
'block_odnopotok_duel_1_1_2', 1, 10);
}

function shutdown_d() {
    apcu_delete($_SERVER['DOCUMENT_ROOT']
'block_odnopotok_duel_1_1_2');
}

$mc->query("LOCK TABLES `user_to_duel` WRITE, `msg`
WRITE, `huntb_list` WRITE, `users` WRITE, `response`
WRITE, `userbag` WRITE, `shop` WRITE, `battle`
WRITE, `userHuntNotActiveMob` WRITE, `shop_equip` WRITE");

$battle_start_time = time();

```

```

    $mc->query("DELETE FROM `userHuntNotActiveMob` WHERE
`time_end` < '$battle_start_time'");
    $mc->query("DELETE FROM `response` WHERE
`time_start` < '$battle_start_time' && `type` != '9'");
    $user_to_duelRes = $mc->query("SELECT * FROM
`user_to_duel` WHERE `unix_time` <
'$battle_start_time'");
    if ($user_to_duelRes->num_rows != 0) {
        $user_to_duel = $user_to_duelRes-
>fetch_all(MYSQLI_ASSOC);
        foreach ($user_to_duel as $value) {
            $mc->query("INSERT INTO `msg`
(`id_user`,`message`,`date`,`type`) VALUES
('${value['user_id1']}', '!','" . time() . "','msg')");
            $mc->query("INSERT INTO `msg`
(`id_user`,`message`,`date`,`type`) VALUES
('${value['user_id2']}', '!','" . time() . "','msg')");
            $mc->query("DELETE FROM `user_to_duel` WHERE
`id` = '${value['id']}'");
        }
    }
    $time_wait = 5;
    $time_out_hero = time() + 600;
    $botNum = 1;
    $fieldUsers =
`side`,`league_exp`,`league`,`level`,`id`,`temp_health`
`,`health`,`strength`,`toch`,`lov`,`kd`,`block`,`breaks`
`hield`,`bron`,`stil`,`name`,`superudar`,`location`,`cri`
`tical`;

```

```

    $arrAllRes = $mc->query("SELECT * FROM `huntb_list`
WHERE `type`='1' || `type`='2'");
    if ($arrAllRes->num_rows != 0) {
        while ($arrNext = $arrAllRes-
>fetch_array(MYSQLI_ASSOC)) {
            $arrNextThisRes = $mc->query("SELECT * FROM
`huntb_list` WHERE `user_id` = '" . $arrNext['user_id']
. "'&& `type`='" . $arrNext['type'] . "'");
            if ($arrNextThisRes->num_rows != 0) {
                $arrNextThis = $arrNextThisRes-
>fetch_array(MYSQLI_ASSOC);
                $userRes = $mc->query("SELECT
$fieldUsers FROM `users` WHERE `id` = '" .
$arrNextThis['user_id'] . "' && `temp_health`>'0' ");
                if ($userRes->num_rows != 0) {
                    $user = $userRes-
>fetch_array(MYSQLI_ASSOC);
                    $battle_id = rand(10, time()) .
rand(10, time()) . rand(10, time());
                    $arrMRes = $mc->query("SELECT
$fieldUsers FROM `users` WHERE `id` IN (SELECT `user_id`
FROM `huntb_list` WHERE `type`='" . $arrNextThis['type']
. "' && `stars` = '" . $arrNextThis['stars'] . "' &&
`user_id` NOT IN (SELECT `id_user2` FROM `response` WHERE
`id_user1` = '" . $user['id'] . "')) && `id` != '" .
$user['id'] . "' && `temp_health`>'0' LIMIT 1");
                    if ($arrMRes->num_rows > 0) {
                        $arrM = $arrMRes-
>fetch_array(MYSQLI_ASSOC);

```

```

        $user2      =      $mc->query("SELECT
$fieldUsers FROM `users` WHERE `id` = '" . $arrM['id'] .
"'")->fetch_array(MYSQLI_ASSOC);
        hero1_add('0',          $user,
$user2['name'],      $battle_id,      $battle_start_time,
$arrNextThis['type']);
        hero2_add('1',          $user2,
$battle_id, $battle_start_time, $arrNextThis['type']);
        $control0 = $mc->query("DELETE
FROM `huntb_list` WHERE `user_id` = '" . $user['id'] .
"'");
        $control1 = $mc->query("DELETE
FROM `huntb_list` WHERE `user_id` = '" . $arrM['id'] .
"'");
        $control3 = $mc->query("INSERT
INTO      `response`(`id`,      `id_user1`,      `id_user2`,
`time_start`, `type`) VALUES ('NULL','" . $user['id'] .
"', '" . $arrM['id'] . " ', '$time_out_hero', '" .
$arrNextThis['type'] . "')");
        $control4 = $mc->query("INSERT
INTO      `response`(`id`,      `id_user1`,      `id_user2`,
`time_start`, `type`) VALUES ('NULL','" . $arrM['id'] .
"', '" . $user['id'] . " ', '$time_out_hero', '" .
$arrNextThis['type'] . "')");

        require_once
$_SERVER['DOCUMENT_ROOT'] . '/functions/phptonode.php';
        $arr      =      ['loadBattle'      =>
$battle_id];

        tonode($arr);

```

```

    } else if
($arrNextThis['time_start'] + $time_wait < time()) {
    $arrbotpar = genbotpar($names,
$botNum);
    herol_add('0', $user,
$arrbotpar[0][0], $battle_id, $battle_start_time,
$arrNextThis['type']);
    for ($i = 0; $i <
count($arrbotpar[0]); $i++) {
        // [BOT]
        bot_add($arrbotpar[0][$i],
'1', $arrbotpar[1][$i], $arrbotpar[2][$i],
$user['level'], $battle_id, $battle_start_time,
$arrNextThis['type']);
    }
    $control2 = $mc->query("DELETE
FROM `huntb_list` WHERE `user_id` = '" . $user['id'] .
"'");
    require_once
$_SERVER['DOCUMENT_ROOT'] . '/functions/phptonode.php';
    $arr = ['loadBattle' =>
$battle_id];
    tonode($arr);
}
}
}
}
}
}

```

```

function herol_add($command, $user, $user2_name,
$battle_id, $battle_start_time, $type) {
    global $mc;
    $arr1 = [[], []];
    $shops_ids = [];
    $PA = $user;
    $PA['weaponico'] = 0;
    $PA['Pshieldnum'] = 0;
    $PweaponEffect = array();

    //pl 1
    $arr1 = [];
    $arr1['temp_health'] = $PA['temp_health'];
    $arr1['max_health'] = $PA['health'];
    $arr1['strength'] = $PA['strength'];
    $arr1['toch'] = $PA['toch'];
    $arr1['lov'] = $PA['lov'];
    $arr1['kd'] = $PA['kd'];
    $arr1['block'] = $PA['block'];
    $arr1['breakshield'] = $PA['breakshield'];
    $arr1['bron'] = $PA['bron'];
    $arr1['critical'] = $PA['critical'];
    $result221 = $mc->query("SELECT `id_shop`,`icon`
FROM `userbag` WHERE `id_user` = '" . $PA['id'] . "' &&
`dress`='1' && `BattleFlag`='1' || `id_user` = '" .
$PA['id'] . "' AND (`id_punct`>'9' && `id_punct` != '14')
&& `BattleFlag`='1'");
    $myrow221 = $result221->fetch_all(MYSQLI_ASSOC);
    for ($i = 0; $i < count($myrow221); $i++) {

```

```

        //read thing
        $result1 = $mc->query("SELECT
`id`,`name`,`health`,`strength`,`toch`,`lov`,`kd`,`block`,`breakshield`,`bron`,`id_punct`,`id_image`,`koll`,`effects`,`critical` FROM `shop` WHERE `id`='\"
$myrow221[$i]['id_shop'] . \"'");
        if ($result1->num_rows > 0) {
            //thing to arr par
            $infoshop = $result1-
>fetch_array(MYSQLI_ASSOC);
            $shops_ids[] =
[addslashes($infoshop['name']), $infoshop['id']];
            $arr1['max_health'] +=
$infoshop['health'];
            $arr1['strength'] +=
$infoshop['strength'];
            $arr1['toch'] += $infoshop['toch'];
            $arr1['lov'] += $infoshop['lov'];
            $arr1['kd'] += $infoshop['kd'];
            $arr1['block'] += $infoshop['block'];
            $arr1['breakshield'] +=
$infoshop['breakshield'];
            $arr1['bron'] += $infoshop['bron'];
            $arr1['critical'] +=
$infoshop['critical'];
            if ((int) $infoshop['id_punct'] == 1) {
                if ($myrow221[$i]['icon'] != 0) {
                    $PA['weaponico'] =
$myrow221[$i]['icon'];
                } else {

```



```

                $PA['weaponico']           =
$infoshop['id_image'];
                }
            }
            if ((int) $infoshop['id_punct'] == 2) {
                $PA['Pshieldnum']         =
$infoshop['koll'];
            }
            if ($PA['stil'] >= 0 && $PA['stil'] <
10) {
                if
(is_array(json_decode_nice($infoshop['effects']))) {
                    $PweaponEffect       =
array_merge($PweaponEffect,
json_decode_nice($infoshop['effects']));
                }
            }
        }
        $mc->query("INSERT INTO `battle`"
            . "("
            . "`id`,`"
            . "`critical`,`"
            . "`Pname`,`"
            . "`Pnamevs`,`"
            . "`Pvsname`,`"
            . "`level`,`"
            . "`Pico`,`"
            . "`Sflife`,`"
            . "`Pflife`,`"

```

- . "`Plife` ,"
- . "`Ptochnost` ,"
- . "`Pblock` ,"
- . "`breakshield` ,"
- . "`Puron` ,"
- . "`Pbronia` ,"
- . "`Poglushenie` ,"
- . "`Puvorot` ,"
- . "`Pweaponico` ,"
- . "`Pshieldnum` ,"
- . "`Pshieldonoff` ,"
- . "`Ptype` ,"
- . "`Pvisible` ,"
- . "`Mvisible` ,"
- . "`Panimation` ,"
- . "`Manimation` ,"
- . "`Phod` ,"
- . "`Phodtime` ,"
- . "`Pauto` ,"
- . "`PAlwaysEffect` ,"
- . "`PeleksirVisible` ,"
- . "`PweaponEffect` ,"
- . "`PentityEffect` ,"
- . "`MentityEffect` ,"
- . "`super` ,"
- . "`Mid` ,"
- . "`location` ,"
- . "`type_battle` ,"
- . "`battle_id` ,"
- . "`battle_start_time` ,"

```

. "`command`,"
. "`lost_mob_id`,"
. "`player_activ`,"
. "`end_battle`,"
. "`counter`,"
. "`stil`,"
. "`shops_ids`"
. ")VALUES ("
. "NULL,"
. "" . $arr1['critical'] . ","
. "" . $PA['name'] . ","
. "" . $PA['name'] . ","
. "$user2_name',"
. "" . $PA['level'] . ","
. "" . $PA['side'] . ","
. "" . $arr1['max_health'] . ","
. "" . $arr1['max_health'] . ","
. "" . $PA['temp_health'] . ","
. "" . $arr1['toch'] . ","
. "" . $arr1['block'] . ","
. "" . $arr1['breakshield'] . ","
. "" . $arr1['strength'] . ","
. "" . $arr1['bron'] . ","
. "" . $arr1['kd'] . ","
. "" . $arr1['lov'] . ","
. "" . $PA['weaponico'] . ","
. "" . $PA['Pshieldnum'] . ","
. "0',"
. "0',"
. "1',"

```

```

        . "'1',"
        . "'0',"
        . "'0',"
        . "'0',"
        . "" . time() . "',"
        . "'0',"
        . "'[]',"
        . "'1',"
        . "" . json_encode($PweaponEffect) .
"'',"

        . "'[]',"
        . "'[]',"
        . "" . $PA['superudar'] . "',"
        . "" . $PA['id'] . "',"
        . "" . $PA['location'] . "',"
        . "$type',"
        . "" . $battle_id . "',"
        . "" . $battle_start_time . "',"
        . "" . $command . "',"
        . "'0',"
        . "'1',"
        . "'0',"
        . "'0',"
        . "" . $PA['stil'] . "',"
        . "" . json_encode($shops_ids,
JSON_UNESCAPED_UNICODE) . ""
        . ")");
    }

```

Додаток Б

Демонстративні матеріали



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ



Кафедра інженерії програмного забезпечення

МАГІСТЕРСЬКА РОБОТА

«РОЗРОБКА АЛГОРИТМУ ПІДБОРУ ПРОТИВНИКА ДЛЯ
МУЛЬТИМЕДІЙНОЇ ОНЛАЙН ГРИ ЖАНРУ MMORPG»

Виконав: студент групи ПДМ-62 Хом'як Дмитро Миколайович

Керівник: к.т.н., доц., Негоденко Олена Василівна

Київ - 2022

МЕТА, ОБ'ЄКТ, ПРЕДМЕТ ДОСЛІДЖЕННЯ

2

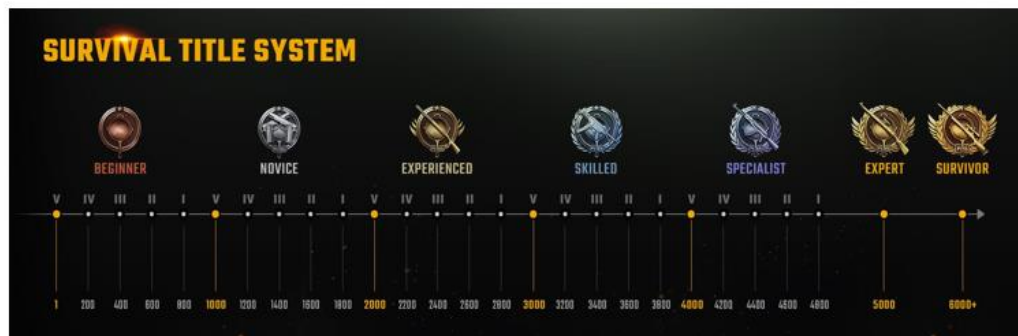
Мета роботи: покращення алгоритму підбору противника в мультиплеєрній онлайн грі

Об'єкт дослідження: процес підбору противника

Предмет дослідження: алгоритми та методи рейтингового підбору противника для гри

АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ЇХ МОДЕЛЕЙ

3



1. Ело-рейтинг.

2. MMR-рейтинг.

Statistics

New ranked matchmaking system

	Herald	Guardian	Cruader	Archon	Legend	Ancient	Divine
0	840	1680	2520	3360	4200	5040	
•	140	980	1820	2660	3500	4340	5180
••	280	1120	1960	2800	3640	4480	5320
•••	420	1260	2100	2940	3780	4620	5460
••••	560	1400	2240	3080	3920	4760	5600
•••••	700	1540	2380	3220	4060	4900	5740

МОДЕЛЬ РЕЙТИНГОВО ПІДБОРУ

4

Рейтинг підбору гравця (MMR – Matchmaking Rating) – це показник, за допомогою якого визначається особистий рейтинг гравця, його навички та вміння грати.

Властивості:

- Кількість отриманих балів за перемогу мінімум в два рази менша за кількість втрачених балів за програв.
- Чим більша загальна кількість балів – тим менша кількість отриманих балів за перемогу. І навпаки, чим менша загальна кількість балів – тим більша кількість отриманих балів за перемогу.
- Захист від втрати балів при певних обставинах. Таких як вихід з гри через поганий інтернет, недобросовісних союзників, які посварились ще до початку гри або раптове вимкнення світла.
- Накопичувальна система кількості перемог або поразок підряд. Тобто, чим більше гравець програє – тим слабших противників йому підбирає. І навпаки, чим більше він приграє – тим сильніших противників йому підбирає.

Математична формула для розрахунку кількості отриманих балів рейтингу при перемозі:

1. E_A - математичне сподівання кількості очок, які набере гравець А проти гравця В.
2. R_A - старий рейтинг гравця А.
3. R_B - старий рейтинг гравця В.

$$E_A = \frac{1}{1 + 10^{\frac{R_B - R_A}{400}}}$$

1. K - коефіцієнт, значення якого дорівнює 10 гравців з рейтингом вище 2400, 15 — для гравців із рейтингом нижче 2400 і 25 — для новачків.
2. S_A - фактично набрана кількість очок гравцем А.
3. R_A - новий рейтинг гравця А.

$$R'_A = R_A + K * (S_A - E_A)$$

РЕЙТИНГ ПЕРСОНАЖА

5

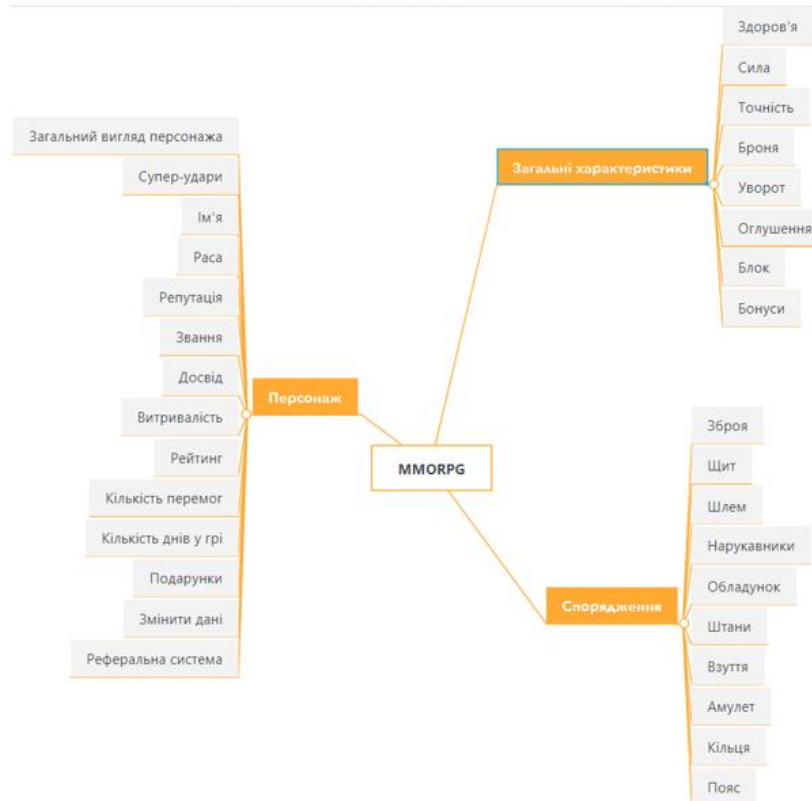
Всі	Рейтинг	Клани
1	HotKot	4802
2	Phora	4582
3	_™ АНАНАС™ _	4574
4	ღღღ ДИВОВИЖНА ღღღ	4505
5	SAYREX	4505
6	Анархія	4489
7	Тренер	4488
8	Вампир	4478

9	@-John сепо-@	4465
10	topollbga56ru	4462
11	Кіт в сапогах	4455
12	Таня	4432
13	Джон Уїк	4414
14	Сахарок	4382
15	Чізі	4365

НАЗАД

ЗАГАЛЬНА СИЛА ПЕРСОНАЖА

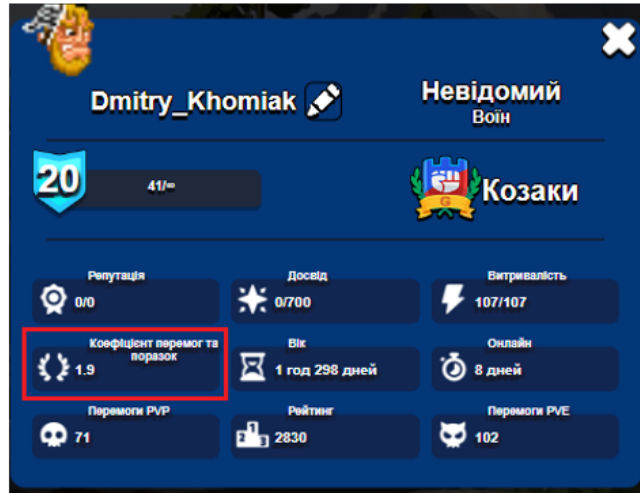
6



КОЕФІЦІЄНТ ПЕРЕМОГ ТА ПОРАЗОК

7

1. Стандартне значення – 1.
2. Крок зміни 0,1 бал в більшу або меншу сторону.
3. Мінімальне значення – 0,1.
4. Максимальне значення не обмежене.



МОДЕЛЬ АЛГОРИТМУ ПІДБОРУ

8



РЕАЛІЗАЦІЯ АЛГОРИТМУ

9

Dmitry_Khomiak [18]			
❤	❤	🔱	👤
2618	2618	225	155
🔱	👤	🔱	👤
276	195	150	227

▶ Переглянути вміст рюкзака

Коефіцієнт підбору: 1951 бал

Viktor_Bender [18]			
❤	❤	🔱	👤
2512	2512	239	188
🔱	👤	🔱	👤
290	173	165	210

▶ Переглянути вміст рюкзака

Коефіцієнт підбору: 2099 балів



РЕАЛІЗАЦІЯ АЛГОРИТМУ

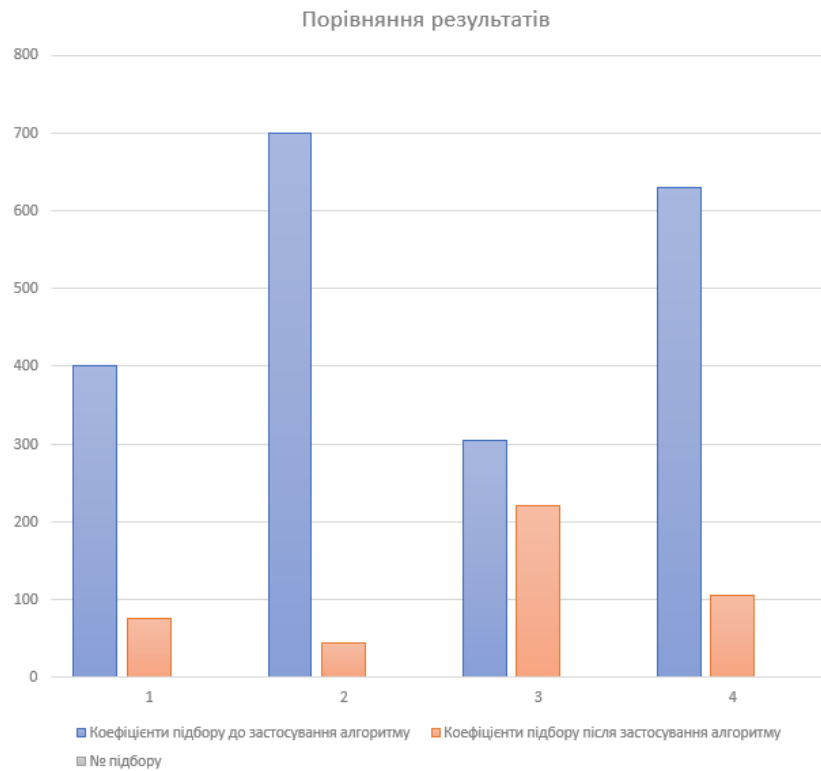
10

Підбір до застосування алгоритму			Підбір після застосування алгоритму		
	Гравець 1	Гравець 2	Підбір 1	Гравець 1	Гравець 2
Рейтинг	2000	2500			2000
Загальна сила персонажа	500	800		500	600
Коефіцієнт перемог та поразок	0,8	1,1		0,8	0,9
Коефіцієнт підбору	400,15		Коефіцієнт підбору	75,05	

Підбір до застосування алгоритму			Підбір після застосування алгоритму		
	Гравець 1	Гравець 2	Підбір 2	Гравець 1	Гравець 2
Рейтинг	3000	2000			3000
Загальна сила персонажа	900	500		900	910
Коефіцієнт перемог та поразок	1,5	1,1		1,5	1,6
Коефіцієнт підбору	700,2		Коефіцієнт підбору	44,95	

	1	2
Похибка із застосуванням алгоритму, %	18,75547	6,4195944
Точність підбору персонажів, %	81,24453	93,5804056
Середнє значення усіх підборів, %	87,30190636	

РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ



ВИСНОВКИ

1. Досліджено ринок комп'ютерних ігор, його основні тенденції як в Україні, так і світі в цілому. Оглянуті найпопулярніші онлайн ігри сьогодення для дослідження їх алгоритмів.
2. Досліджено алгоритми, їхню загальну характеристику, наведений приклад алгоритму підбору.
3. Розглянуті моделі персонажа та рюкзака. Їх основні властивості та характеристики. Досліджений та розроблений алгоритм для знаходження сили використовуючи дані моделі.
4. Спроектований та розроблений алгоритм підбору противника.
5. Проведено апробацію розробленого алгоритму.

ПУБЛІКАЦІ ТА АПРОБАЦІЯ РОБОТИ

13

Тези доповідей:

1. Хом'як Д. М. Розробка алгоритму підбору супротивника в мультиплеерній онлайн грі // III Науково-практична конференція «Проблеми комп'ютерної інженерії» 1 грудня 2022 року, Державний університет телекомунікації, Київ, Україна, – с. 270.

14

ДЯКУЮ ЗА УВАГУ!