МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. Сікорського

Кафедра

інформатики та програмної інженерії

(повна назва кафедри, циклової комісії)

**КУРСОВА РОБОТА**

з Основ програмування. Частина 2. Модульне програмування.

(назва дисципліни)

на тему: «Пекмен»

Студента 1 курсу, групи ІП-21

Загребельного Олександра Андрійовича

Спеціальності 121 «Інженерія програмного забезпечення»

Керівник

Асистент Вовк. Є. А.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Кількість балів: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Національна оцінка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| Члени комісії |  |  | ст. вик. Головченко М. М. |
|  | (підпис) |  | (посада, вчене звання, науковий ступінь, прізвище та ініціали) |
|  |  |  | к.т.н., доц. Муха І. П. |
|  | (підпис) |  | (посада, вчене звання, науковий ступінь, прізвище та ініціали) |

Київ - 2023 рік

КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. Сікорського

(назва вищого навчального закладу)

Кафедра інформатики та програмної інженерії

Дисципліна Основи програмування

Напрям "ІПЗ"

Курс 1 Група ІП-21 Семестр 2

ЗАВДАННЯ

на курсову роботу студента

Загребельного Олександра Андрійовича

(прізвище, ім’я, по батькові)

1. Тема роботи «Гра Пекмен»

2. Строк здачі студентом закінченої роботи 02.06.2023

3. Вихідні дані до роботи Технічне завдання Додаток А.

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

Вступ, постановка задачі, теоретичні відомості, опис алгоритмів, опис програмного

забезпечення, тестування програмного забезпечення, інструкція користувача, висновок.

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

6. Дата видачі завдання 21.03.2023

КАЛЕНДАРНИЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Назва етапів курсової роботи | Термін виконання етапів роботи | Підписи керівника, студента |
| 1. | Отримання теми курсової роботи | 21.03.2023 |  |
| 2. | Підготовка ТЗ | .2023 |  |
| 3. | Пошук та вивчення літератури з питань курсової роботи | .2023 |  |
| 4. | Розробка сценарію роботи програми | .2023 |  |
| 6. | Узгодження сценарію роботи програми з керівником | .2023 |  |
| 5. | Розробка (вибір) алгоритму рішення задачі | .2023 |  |
| 6. | Узгодження алгоритму з керівником | .2023 |  |
| 7. | Узгодження з керівником інтерфейсу користувача | .2023 |  |
| 8. | Розробка програмного забезпечення | .2023 |  |
| 9. | Налагодження розрахункової частини програми | .2023 |  |
| 10. | Розробка та налагодження інтерфейсної частини програми | .2023 |  |
| 11. | Узгодження з керівником набору тестів для контрольного прикладу | .2023 |  |
| 12. | Тестування програми | .2023 |  |
| 13. | Підготовка пояснювальної записки | .2023 |  |
| 14. | Здача курсової роботи на перевірку | 02.06.2023 |  |
| 15. | Захист курсової роботи | 05.06.2023 |  |

Студент Загребельний О. А.

(підпис)

Керівник Вовк. Є. А,

(підпис) (прізвище, ім’я, по батькові)

" 23 " березня 2023 р.

АНОТАЦІЯ

Пояснювальна записка до курсової роботи: 55 сторінок, 24 рисунки, 28 таблиць, 4 посилання.

Мета роботи: розробка програми-аналога відомої гри «Pacman» з використанням принципів об’єктно-орієнтованого програмування.

У результаті виконання роботи були вивчені методи роботи з інтерфейсом за допомогою функціоналу бібліотеки Pygame. Також були розглянуті алгоритми реалізації руху привидів відповідно до їх поведінки на ігровому полі.

Як наслідок, було розроблено програмний продукт, який демонструє реалізацію гри «Пекмен» з використанням принципів об’єктно-орієнтованого програмування.

ПЕКМЕН, ПРИВИДИ, БОНУСИ, ЇЖА, ДВЕРІ, ГРАФІЧНИЙ ІНТЕРФЕЙС, ІГРОВЕ ПОЛЕ, ФАЙЛИ, ОЧКИ, ЛАБІРИНТ, УПРАВЛІННЯ ПЕРСОНАЖЕМ, АЛГОРИТМИ РУХУ ТА ВИБОРУ ЦІЛІ.

Зміст

[Вступ 5](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587135)

[1 Постановка задачі 6](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587136)

[2 Теоретичні відомості 7](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587137)

[3 Опис алгоритмів 8](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587138)

[3.1. Загальний алгоритм 8](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587139)

[3.2. Алгоритм методу 1 8](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587140)

[4 Опис програмного забезпечення 9](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587141)

[4.1. Діаграма класів програмного забезпечення 9](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587142)

[4.2. Опис методів частин програмного забезпечення 9](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587143)

[4.2.1. Користувацькі методи 9](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587144)

[4.2.2. Стандартні методи 9](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587145)

[5 Тестування програмного забезпечення 10](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587146)

[5.1. План тестування 10](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587147)

[5.2. Приклади тестування 10](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587148)

[6 Інструкція користувача 11](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587149)

[7 Аналіз результатів 12](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587150)

[Висновки 13](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587151)

[Перелік посилань 14](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587152)

[Додаток А Технічне завдання 15](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587153)

[Додаток Б Тексти програмного коду 18](file:///C:\Users\zagrebelnio\Desktop\pacman-game\Розділи%20курсової\Курсова_шаблон_2023.docx#_Toc135587154)

Вступ

Гра «Пекмен» є відомою аркадною грою, що досі зберігає свою популярність серед гравців. Вона випереджала свій час, започаткувавши новий жанр ігор та встановивши стандарти для майбутніх ігрових розробок. Одним із завдань курсової роботи є дослідження ігрових механік і алгоритмів, які використовуються в оригінальній грі, та розробка власного програмного забезпечення на основі них.

Актуальність роботи полягає в тому, що вона надає можливість вивчити та застосувати проаналізовані механіки у контексті створення ігрових проектів. Гра «Пекмен» демонструє широкий спектр концепцій, таких як рух персонажів, алгоритми пошуку шляху та прийняття рішень.

Загальною ціллю курсової роботи є розвиток та покращення навичок проектування і реалізації програмного забезпечення для поставлених глобальних задач та проектів, застосування набутих знань при дослідженні предметної області на практиці та їх закріплення.

# Постановка задачі

Розробити програмне забезпечення для гри «Пекмен», яке включає наступні функціональні можливості:

1. Створення гри:

* Розміщення пекмена та привидів на ігровому полі.
* Розміщення їжі та бонусів на ігровому полі.

1. Рух та взаємодія об'єктів:

* Забезпечення руху пекмена по лабіринту, включаючи можливість зміни напрямку руху.
* Рух ворогів згідно з алгоритмами їхньої поведінки.
* Виявлення колізій між об'єктами (наприклад, колізія пекмена з їжею або привидом).

1. Логіка гри:

* Підрахунок очок за збирання кульок.
* Визначення умови перемоги (збирання усієї їжі) та умови поразки (зіткнення з привидами тричі).
* Відображення результатів гри (кількість очок).

1. Взаємодія з користувачем:

* Графічний інтерфейс користувача для відображення гри та взаємодії з нею (відображення лабіринту, об'єктів і результатів).
* Управління пекменом за допомогою клавіатури.

Вхідними даними для програми є параметри налаштування гри, такі як матриця лабіринту, його розміри лабіринту та розміщення об'єктів. Вихідними даними є результати гри (перемога чи поразка) та кількість набраних очок.

Опис правил: задано ігрове поле, на якому наявні стіни лабіринту, їжа, чотири бонуси та привиди, пекмен, двері та інтерфейс відображення набраних очок та життів гравця. Завдання гравця – зібрати усю їжу та бонуси. На заваді йому стоять чотири привиди, при зіткненні з якими гравець «помирає». У разі смерті пекмена він втрачає одне з трьох життів, які даються йому на початку гри. У разі втрати усіх життів гравець програє, натомість якщо йому вдалося зібрати усю їжу на ігровому полі, гра вважається виграною. На початку гри та після кожної смерті гравця усі привиди знаходяться у своїй зачиненій кімнаті, яка відчиняється через 3 секунди після початку руху гравця. Приклад початкового стану ігрового поля можна побачити на (Рисунку 1.1).

Зображення, що містить знімок екрана

Автоматично згенерований опис

Рисунок 1.1 – Ігрове поле

# Теоретичні відомості

2.1. Лабіринт та його структура

Лабіринт у грі «Пекмен» складається з комірок, де деякі комірки можуть бути зайняті стінами, а інші доступні для руху. Це впливає на можливість переміщення об'єктів та визначення їх шляхів.

Таким чином ігрове поле можна представити числовою матрицею, де кожен елемент матриці відповідає за його наповнення:

-1 – порожня комірка;

0 – комірка зі звичайною іжею;

1 – комірка зі спеціальним бонусом;

2 – комірка з дверима для привидів;

3-15 – комірки зі стінами;

2.2. Координати та рух об'єктів

У грі Pacman лабіринт представлений у вигляді двовимірної сітки, де кожна комірка має свої координати (x, y). Об'єкти, такі як пекмен і привиди, також знаходяться у певних координатах на цій сітці. Рух об'єктів відбувається шляхом зміни їх координат залежно від введення користувача або алгоритмів руху.

2.3. Рух об'єктів

Пекмен та привиди рухаються по лабіринту, дотримуючись певних правил. Пекмен може змінювати свій напрямок руху в залежності від вибору користувача однієї з доступних сусідніх комірок. Кожен раз, коли користувач бажає змінити напрямок руху пекмена, перевіряється відповідна комірка по відношенню до поточної клітинки пекмена. Якщо сусідня комірка стіна, туди повернути не можна, якщо ні, то пекмен змінить напрямок руху. Вороги також рухаються, але за визначеними алгоритмами в залежності від їх типу та завдання (патрулювання області, переслідування гравця, охорона бонусів).

Загалом на ігровому полі є чотири привиди: два привиди-переслідувачі, привид-охоронець та привид-патрульний.

У кожного з них є комірка-ціль, до якої привид прямує. Різниця між ними полягає в тому, як ця комірка визначається. Наприклад, привиди-переслідувачі орієнтуються на комірку, в якій знаходиться пекмен. Для першого з них коміркою-ціллю є безпосередньо клітинка, в якій на момент гри знаходиться пекмен, а для другого клітинка, яка знаходиться на дві попереду відносно напрямку пекмена. Комірка-ціль для привида-охоронця – випадково обрана клітинка одного з чотирьох бонусів. Якщо пекмен збере бонус, який був ціллю для цього привида, він обере іншу комірку-бонус, а у разі, коли їх не залишиться, його ціль залишиться останньою, що була для нього актуальною. Привид-патрульний має дві програмно задані патрульні області на ігровому полі. На початку гри обирається одна із цих двох областей, які він буде патрулювати протягом усієї гри. Після задання області для цього привида обирається комірка-ціль в межах цієї області, яка не може бути стіною. При досягненні цієї комірки привидом вона переобирається в заданій області. Приклад цілей для привидів під час гри можна побачити на (Рисунку 2.1).

Зображення, що містить знімок екрана

Автоматично згенерований опис

Рисунок 2.1 – Комірки-цілі для привидів

2.4. Зіткнення та взаємодія об'єктів

У грі «Пекмен» можуть відбуватись зіткнення між об'єктами, наприклад, пекмен може зіткнутись зі стіною, ворогом, їжею або бонусом. Для цього під час ігрового процесу перевіряються колізії між різними об’єктами.  
 Колізія пекмен-стіна: перевіряються чотири види колізій (пекмен зіштовхується зі стіною, рухаючись праворуч, ліворуч, вгору або вниз). На усі чотири випадки перевіряються умови відношення координат об’єктів. Наприклад, для зіткнення справа перевіряється чи крайня права частина пекмена знаходиться в межах блоку стіни. Якщо це так, то координата x центру пекмена перевизначається таким чином:

, де:

координата x центру пекмена;

координата x лівого краю блоку;

радіус пекмена;

Також взаємодія об'єктів включає збирання їжі та бонусів пекменом, що збільшує його кількість очок. Колізія пекмена з їжею визначається за відношенням координат їх центрів та радіусів за наступною формулою:

Якщо обидві нервіності виконуються, значить колізія між пекменом та їжею відбулася, а отже відповідний об’єкт їжі видаляється, а пекмену нараховуються очки.

2.5. Правила нарахування балів та умови завершення гри

Загальна кількість очок залежить від результату гри, кількості смертей та часу, витраченого на гру. Якщо гра закінчилась поразкою гравця, то кількість очок залишається незмінною. У разі перемоги рахунок гравця домножується на спеціальні коефіцієнти, які залежать від кількості смертей пекмена за всю гру та час, витрачений на її проходження. Залежність коефіцієнтів від результатів гри наведено в таблиці 2.1.

Таблиця 2.1 – коефіцієнти-множники очок

|  |  |  |
| --- | --- | --- |
| Статус гри | | Множник |
| Результат | Кількість смертей |
| Поразка | 3 | 1 |
| Перемога | 2 | 1.5\*k(t) |
| Перемога | 1 | 2\*k(t) |
| Перемога | 0 | 3\*k(t) |

k(t) – це коефіцієнт, який залежить від часу (в секундах), витраченого на проходження гри.

Таким чином, якщо гравець впорався за 5 або менше хвилин, його остаточні бали помножаться на відповідний коефіцієнт. Якщо знадобилося більше часу, коефіцієнт дорівнюватиме 1.

# Опис алгоритмів

Таблиця 3.1 – Основні змінні та їх призначення

|  |  |
| --- | --- |
| Змінна | Призначення |
| blocks | Матриця координат, ширини та довжини блоків стін |
| direction | Напрямок об’єкта |

## Загальний алгоритм

1. ПОЧАТОК

2. Зчитати розмірність лабіринту.

3. Згенерувати початкову конфігурацію лабіринту з об'єктами (стіни, їжа, привиди, тощо).

4. Вивести початковий стан гри.

5. Ігровий цикл допоки гра триває

5.1. Отримати дії гравця

5.2. Виконати поворот гравця залежно від отриманих дій. (3.2)

5.3. Перевірити зіткнення зі стінами (3.3)

5.4. Оновити бали та статистику гри.

5.5. Вивести оновлений стан гри.

6. Вивести результати гри та записати їх у файл (кількість набраних балів).

7. КІНЕЦЬ

3.2. Поворот пекмена

1 ПОЧАТОК

2 Перевірити можливість повороту:

2.1 ЯКЩО натиснуто ліву стрілку, ТО перевірити можливість повороту ліворуч:

2.1.1 ЯКЩО клітинка ліворуч не стіна І координата y пекмена знаходиться в центрі клітинки, ТО присвоїти direction значення “left”.

2.2 ІНАКШЕ ЯКЩО натиснуто праву стрілку, ТО перевірити можливість повороту праворуч:

2.2.1 ЯКЩО клітинка праворуч не стіна І координата y пекмена знаходиться в центрі клітинки, ТО присвоїти direction значення “right”.

2.3 ІНАКШЕ ЯКЩО натиснуто стрілку вниз, ТО перевірити можливість повороту вниз:

2.3.1 ЯКЩО клітинка праворуч не стіна І координата y пекмена знаходиться в центрі клітинки, ТО присвоїти direction значення “right”.

2.4 ІНАКШЕ ЯКЩО натиснуто стрілку вгору, ТО

## 3.3. Зіткнення зі стінами

1 ПОЧАТОК

2 Перевірити зіткнення об’єкту зі стіною:

2.1 ЦИКЛ проходу по всіх рядках матриці blocks.

2.1.1. ЯКЩО напрямок об’єкта праворуч І крайня права межа об’єкта знаходиться в межах блоку, ТО присвоїти координаті x об’єкта значення x лівої межі блоку мінус радіус об’єкта.

2.1.2 ІНАКШЕ ЯКЩО напрямок об’єкта ліворуч І ліва межа об’єкта знаходиться в межах блоку, ТО присвоїти координаті x об’єкта значення x правої межі блоку плюс радіус об’єкта.

2.1.3 ІНАКШЕ ЯКЩО напрямок об’єкта вгору І верхня межа об’єкта знаходиться в межах блоку, ТО присвоїти координаті y об’єкта значення y нижньої межі блоку плюс радіус об’єкта.

2.1.2 ІНАКШЕ ЯКЩО напрямок об’єкта вниз І нижня межа об’єкта знаходиться в межах блоку, ТО присвоїти координаті y об’єкта значення y верхньої межі блоку мінус радіус об’єкта.

3 КІНЕЦЬ

# Опис програмного забезпечення

## Діаграма класів програмного забезпечення

## Опис методів частин програмного забезпечення

### Користувацькі методи

### Стандартні методи

# Тестування програмного забезпечення

## План тестування

## Приклади тестування

# Інструкція користувача

# Аналіз результатів

Висновки

Перелік посилань

Додаток А Технічне завдання

КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. Сікорського

Кафедра

інформатики та програмної інженерії

Затвердив

Керівник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_201\_ р.

Виконавець:

Студент \_\_\_\_<*ПІБ>\_\_\_\_\_\_\_\_\_\_*

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_201\_ р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання курсової роботи

на тему: <*Тема курсової роботи*>

з дисципліни:

«Основи програмування»

Київ 202\_

* 1. *Мета*: Метою курсової роботи є …………..………..………………
  2. *Дата початку роботи*: «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_202\_ р.
  3. *Дата закінчення роботи*: «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 202\_ р.
  4. *Вимоги до програмного забезпечення*.

1. Функціональні вимоги:

* Можливість ……………………………………………………………….
* < *вимоги до функціональних характеристик*>.

1. Нефункціональні вимоги:

* Можливість ……………………………………………………………….
* <*вимоги до надійності, складу і параметрів технічних засобів тощо*>
* Все програмне забезпечення та супроводжуюча технічна документація повинні задовольняти наступним ДЕСТам:

ГОСТ 29.401 - 78 - Текст програми. Вимоги до змісту та оформлення.

ГОСТ 19.106 - 78 - Вимоги до програмної документації.

ГОСТ 7.1 - 84 та ДСТУ 3008 - 2015 - Розробка технічної документації.

* 1. *Стадії та етапи розробки*:

1. Об'єктно-орієнтований аналіз предметної області задачі (до\_\_.\_\_.202\_ р.)
2. Об'єктно-орієнтоване проектування архітектури програмної системи (до \_\_.\_\_.202\_р.)
3. Розробка програмного забезпечення (до \_\_.\_\_.202\_р.)
4. Тестування розробленої програми (до \_\_.\_\_.202\_р.)
5. Розробка пояснювальної записки (до \_\_.\_\_.202\_ р.).
6. Захист курсової роботи (до \_\_.\_\_.202\_ р.).
   1. *Порядок контролю та приймання*. Поточні результати роботи над КР регулярно демонструються викладачу. Своєчасність виконання основних етапів графіку підготовки роботи впливає на оцінку за КР відповідно до критеріїв оцінювання.

Додаток Б Тексти програмного коду

*студента групи ІП-ХХ І курсу*

*Гуменського В.Л.*

(Найменування програми (документа))

*Тексти програмного коду програмного забезпечення вирішення задачі обернення матриць*

(Вид носія даних)

*CD-RW*

(Обсяг програми (документа), арк., Кб)

*182 арк, 124 Кб*