Міні-проект Donut Dream

1) Опис гри

Події гри Donut Dream відбуваються частково в реальному світі та у сні головного героя. Розпочинається гра проханням ввести гравця його нікнейм, після чого розпочинається перший акт(Всього їх три)

Перший акт

Гра розпочинається і головний герой розуміє, що у нього закінчилось усе солодке, а особливо пончики, які він обожнює. У гравця з'являється вибір: або піти сумним та розбитим спати, або піти в магазин та купити щось солоденьке. Якщо гравець іде купляти щось солодке то він може або вдягнути куртку, або ні і втратити 10/100 здоров'я. Пізніше, гравець отримує вибір що купити в магазині: пончик або шоколадку. Залежно від цього вибору він отримує більше або менше додаткових очків. Гравець повертається додому, і з'являється перша міні-гра, де йому потрібно перевести цифри з бінарних в звичайні щоб потрапити до себе в кімнату за три спроби, після чого код змінюється.

Другий акт

Гравець засинає і бачить перед собою розбиту та наполовину з'їджену вафлю в уявному світі здорової їжі. У гравця є 4 вибори: або поговорити з нею, або не реагувати на неї, або вбити її. Усі варіанти вибору приводять героя до замку та до фінального боса, проте якщо гравець обирає вбити вафлю то він отримує або додатковий урон або збільшує собі максимальну кількість здоров'я, втративши при цьому якусь його частину під час битви. Якщо гравець не вибирає вбити вафлю, то головний герой отримує колу, яка регенерує 25 здоров'я гравця, зразу опісля гра рекомендує гравцеві її використати, адже гравець падає з обриву і втрачає 50 очків здоров'я. Гравець може не використовувати її, тоді в кінці гри, якщо гравець виживе, йому будуть нараховані додаткові 200 очків

Третій акт(бос)

Усі варіанти вибору гравця приводять його до замку, де він буде змагатись з першим босом. У нього є 100 здоров'я та його урон — 30.

Гравець може частково відбивати атаку, якщо він відгадає, яке число було помножене на дане число щоб вийшло показане число за 3 спроби.

Гравець може нанести у два рази більше урону якщо відгадає випадкове число від 0 до 150 за 10 спроб. Після кожної спроби гра буде підказувати гравцю чи менше чи більше загадане число від введеного

Після кожних двох атак та захистів, у гравця є можливість відновити частково своє здоров'я, вгадавши відповідь до легеньких загадок. Гравець відновлює або 25 або 50 здоров'я, залежно від того, що йому випаде.

Уся битва з босом знаходиться в циклі, який переривається або коли гравець помер, або коли гравець вбив боса.

2) Опис того, як працює гра, приклади

Graphical user interface, text, application

Description automatically generated

Гравця зустрічає даний інтерфейс та прохання ввести нікнейм

Text

Description automatically generated

Ось, наприклад, вибір гравцем дії. Якщо гравець не ввів нічого або вводить не варіанти відповіді, а текст, то вивидиться поле, яке просить ввести варіант ще раз

Text

Description automatically generated

Приклад гри з відгадуванням паролю до дверей, переводячи бінарні числа в натуральні.

Text

Description automatically generated

Після кожного з трьох актів гравцю виводиться його статистика

Text

Description automatically generated

Міні-гра під час битви з босом на вгадування числа на яке було помножене число

Text

Description automatically generated

Приклад міні-гри з відгадуванням числа від 0 до 150

Text

Description automatically generated

Приклад загадки для відновлення здоров'я

Text

Description automatically generated

Консоль після виграшу гравця

3) Процес розробки

Розпочалось все з ідеї — зробити щось незвичне. Коли я придумував дану ідею, я дуже сильно хотів чогось солоденького, і мені чомусь на думку спали пончики. І далі я продовжив ідею, а події відбуваються в основному уві сні, тому що я в той час дуже хотів спати.

Потім я почав малювати маркерами на дошці у себе в кімнаті структуру та пізніше переносив все це в нотатки, насичуючи репліками, а опісля переносив на python.

І в кінці-кінців було тестування гри, під час якого було виправлено багато помилок та багів.

4) Опис структури гри

У грі наявні 10 функцій, 3 з яких є вкладеними в іншій. Розпочинається код ф-цією start\_game(). У ній гравець вводить свій нікнейм, а система впевнюється, що ім'я було введене.

Наступна функція — rand\_num\_game(). Функція генерує випадковий пароль, де кожна цифра — двійковий бінарний код. Гравцеві треба відгадати пароль, написавши кожну цифру через пробіл.

Бінарний код — код, де кожна наступна цифра(0 чи 1) починаючи справа показує, чи добавляти до числа два в степені починаючи з нуля і добавляючи один до степеня з кожним кроком вправо. 1 — добавляти, 0 — ні. У гравця є три спроби, після чого пароль змінюється.

Функція first\_act(health, points, nickname). Ця функція представляє гравцю сюжет першої глави гри, приймає нік, здоров'я та очки гравця, повертає після закінчання своєї роботи здоров'я та очки гравця.

Функція second\_act(max\_health, health, damage, points, coke). Функція представляє гравцю продовження сюжету гри, приймає значення статистик гравця(урон, максимальна к-сть здоров'я та саме здоров'я, очки), а також таку змінну як coke. В один момент гри гравець отримує колу, яка може добавити здоров'я гравцю, або він може зберегти її, за що в кінці гри він отримає додаткові бали. Функція змінює статистики гравця залежно від того, який вибір він робив та повертає статистику.

Функція the\_boss(max\_health, health, damage, points). Функція приймає статистику гравця та представляє битву з босом гравцю. Якщо гравець перемагає, то він отримує виграшний екран, привітання з перемогою та кількість зароблених очків. Якщо ж гравець помирає, то він отримує кількість очків та гра зупиняється.

Усередині функції the\_boss() є три функції, які забезпечують якісний та безперирвний босфайт.

Функція your\_damage(is\_more\_damage) дає гравцеві можливість відгадати загадане число від 0 до 150 за 10 спроб і при успішному відгадуванні числа ф-ція повертає значання True в is\_more\_damage, що дозволяє гравцю нанести більше урону. Якщо не відгадав, то False

Функція reflection(is\_active\_sheild) дає гравцю можливість відгадати, на яке число було помножене дане, щоб отримати певний результат за три спроби. У разі успіху, головний герой отримує менше урону, тобто функція повертає значення True. Якщо не відгадав, то False

Функція health\_boost(health) дозволяє гравцю під час битви відновити певну кількість здоров'я, якщо гравець відгадає з першої спроби відповідь на загадку(True or False). Вона випадковим чином після успішного відгадування загадки повертає або coke(+ 25 здоров'я) або pepsi(+ 50 здоров'я). Якщо гравець неправильно відповів на загадку — додаткового здоров'я не отримує. Функція повертає значення здоров'я(health)

Функція results(points, max\_player\_health, player\_health, player\_damage) функція приймає значення статистик гравця та виводить їх в кінці кожного акту

Функція main() запускає всі функції у правильному порядку

У грі використовується велика кількість циклів та if-else функцій. В основному використовується цикл while при кожній можливості гравця вибрати варіант продовження гри, для отримання від гравця правильного вибору, а не рандомного набраного тексту, також while використовується під час фінальної битви з босом. Цикли for використовуються, коли потрібно порахувати, чи використав користувач усі спроби під час міні-ігор, якщо їх у нього кінечна кількість