**Background Story:** In this game, players dive into the heart of Gotham City's mysteries as Martian Manhunter and they investigate the disappearance of Commissioner Gordon. The journey begins with crucial choices: collaborate with colleagues or pursue the investigation alone. From there, players embark on a thrilling adventure, exploring Arkham Asylum, Wayne Manor, and the streets of Gotham, encountering Batman and the Justice League along the way. Key moments include discovering a spell book, unraveling the truth behind Gordon's disappearance, and confronting challenging dilemmas. Ultimately, players navigate through multiple endings, from success to failure, culminating in a gripping conclusion that reflects their choices and actions throughout the game.

**Short Overview:**

Investigate Commissioner Gordon's disappearance in Gotham City as Martian Manhunter, encountering Batman, the Justice League, and unraveling the truth through pivotal choices. Navigate through multiple endings, from success to failure, shaping the fate of Gotham in this thrilling adventure.

**Pseudocode:**

Function intro():

Display message to introduce the player about Gotham City and the situation.

Wait for 6 seconds

Display mission details about uncovering Commissioner Gordon's disappearance.

Wait for 5 seconds

Display instructions for the next step.

Wait for 3 seconds

Prompt user to press Enter to start investigation.

Call investigate\_police\_department()

End function

import time

function intro():

print("Hello Green Martian")

time.sleep(3)

print("Welcome to Gotham City...")

time.sleep(6)

print("Your mission is to unravel...")

time.sleep(5)

print("Your next step is to investigate...")

time.sleep(3)

input("Press Enter to begin your investigation...")

investigate\_police\_department()

function continue\_or\_exit():

loop:

choice = input("Do you want to continue playing? (yes/no): ")

if choice is "yes":

continue\_game()

break

else if choice is "no":

print("Thank you for playing! Goodbye.")

break

else:

print("Invalid input. Please enter 'yes' to continue or 'no' to exit.")

function investigate\_police\_department():

print("As you step into the Gotham City Police Department...")

time.sleep(4)

print("What will you do?")

print("1. Search Commissioner Gordon's office for clues.")

print("2. Speak with officers and gather information.")

decision = input("Enter your choice (1/2): ")

if decision is "1":

outcome\_a1()

else if decision is "2":

outcome\_b1()

else:

print("Invalid choice. Please try again.")

investigate\_police\_department()

function outcome\_a1():

print("Carefully searching through Commissioner Gordon's office...")

time.sleep(4)

print("What will you do?")

print("1. Attempt to decrypt the files.")

print("2. Leave them for now and search elsewhere.")

print("3. ShapeShift into Commissioner Gordon to unlock the encrypted files.")

decision = input("Enter your choice (1/2/3): ")

if decision is "1":

decrypt\_files()

else if decision is "2":

print("You decide to leave the files for now and search for other leads.")

outcome\_b1()

else if decision is "3":

print("You shapeshifted into Commissioner Gordon in the corner of his room.")

outcome\_b2()

else:

print("Invalid choice. Please try again.")

outcome\_a1()

function decrypt\_files():

print("You begin the decryption process...")

time.sleep(4)

print("What will you do?")

print("1. Continue decrypting the files, risking exposure.")

print("2. Leave the files for now and search elsewhere.")

decision = input("Enter your choice (1/2): ")

if decision is "1":

continue\_decryption()

else if decision is "2":

print("You reluctantly leave the files behind...")

outcome\_b1()

else:

print("Invalid choice. Please try again.")

decrypt\_files()

function continue\_decryption():

print("Hours pass as you tirelessly work to decrypt the files...")

time.sleep(4)

print("You've cracked the encryption!...")

time.sleep(4)

outcome\_a1\_1()

function outcome\_a1\_1():

print("Among the decrypted files...")

print("What will you do next?")

print("1. Share the evidence with Batman for further analysis.")

print("2. Keep the evidence and continue the investigation alone.")

decision = input("Enter your choice (1/2): ")

if decision is "1":

print("You decide to share the evidence with Batman...")

continue\_or\_exit()

else if decision is "2":

print("You choose to keep the evidence...")

continue\_or\_exit()

else:

print("Invalid choice. Please try again.")

outcome\_a1\_1()

function outcome\_b1():

print("You approach a group of officers...")

time.sleep(3)

print("What will you do?")

print("1. Press the officers for information, using your authority.")

print("2. Try to earn their trust before asking any questions.")

decision = input("Enter your choice (1/2): ")

if decision is "1":

print("You adopt a firm tone...")

game\_over()

else if decision is "2":

print("Recognizing the officers' apprehension...")

continue\_or\_exit()

else:

print("Invalid choice. Please try again.")

outcome\_b1()

function outcome\_b2():

print("You Shapeshift into Commissioner Gordon...")

time.sleep(3)

print("What will you do?")

print("1. Kill the guards")

print("2. Try to earn their trust by stating you are Commissioner Gordon.")

print("3. Flee from the scene")

print("4. Enter their body to erase their memory.")

decision = input("Enter your choice (1/2/3/4): ")

if decision is "1":

game\_over\_worst\_ending()

else if decision is "2":

continue\_or\_exit2()

else if decision is "3":

game\_over()

else if decision is "4":

continue\_or\_exit3()

else:

print("Invalid choice. Please try again.")

outcome\_b2()

function continue\_game():

Display continuation message and options for the next move

Accept user input for the next move

If input is "1":

Call investigate\_arkham\_asylum()

Else if input is "2":

Call explore\_wayne\_manor()

Else if input is "3":

Call patrol\_gotham\_streets()

Else:

Display error message and prompt user to try again

Recursively call continue\_game()

function continue\_game2():

Display continuation message and options for the next move

Accept user input for the next move

If input is "1":

Call investigate\_arkham\_asylum\_with\_justice\_league()

Else if input is "2":

Call explore\_wayne\_manor\_with\_justice\_league()

Else if input is "3":

Call patrol\_gotham\_streets()

Else:

Display error message and prompt user to try again

Recursively call continue\_game2()

function continue\_game3():

Display continuation message and options for the next move

Accept user input for the next move

If input is "1":

Call game\_over\_guilty\_ending()

Else if input is "2":

Call patrol\_gotham\_streets()

Else:

Display error message and prompt user to try again

Recursively call continue\_game3()

function investigate\_arkham\_asylum():

Display introduction to Arkham Asylum and options for investigation

Accept user input for investigation choice

If input is "1":

Proceed with speaking with staff and inmates

Else if input is "2":

Proceed with searching the asylum's records

Else:

Display error message and prompt user to try again

Recursively call investigate\_arkham\_asylum()

function investigate\_arkham\_asylum\_with\_justice\_league():

Display introduction to Arkham Asylum with the Justice League and options for investigation

Accept user input for investigation choice

If input is "1":

Proceed with speaking with staff and inmates

Else if input is "2":

Proceed with searching the asylum's records

Else if input is "3":

Proceed with questioning Riddler

Else:

Display error message and prompt user to try again

Recursively call investigate\_arkham\_asylum\_with\_justice\_league()

function explore\_wayne\_manor():

Display introduction to Wayne Manor and options for exploration

Accept user input for exploration choice

If input is "1":

Proceed with searching for clues in the Wayne family archives

Else if input is "2":

Proceed with exploring secret passages and hidden chambers

Else:

Display error message and prompt user to try again

Recursively call explore\_wayne\_manor()

function patrol\_gotham\_streets():

Display introduction to patrolling Gotham streets and options for patrol

Accept user input for patrol choice

If input is "1":

Proceed with listening for rumors and whispers

Else if input is "2":

Proceed with investigating recent disturbances

Else:

Display error message and prompt user to try again

Recursively call patrol\_gotham\_streets()

function explore\_wayne\_manor\_with\_justice\_league():

Display information about Batman's parent's death anniversary and options for action

Accept user input for action choice

If input is "1":

Proceed with apologizing to Batman and continuing investigation

Else if input is "2":

Proceed with telling Batman about the spellbook

Else:

Display error message and prompt user to try again

Recursively call explore\_wayne\_manor\_with\_justice\_league()

function joker():

Display encounter with Joker and options for action

Accept user input for action choice

If input is "1":

Proceed with killing Joker

Else if input is "2":

Proceed with getting in Joker's body to read his mind

Else if input is "3":

Proceed with taking the book from Joker and informing Batman

Else:

Display error message and prompt user to try again

Recursively call joker()

# Game Over Functions

function game\_over():

Display message indicating game over and reason for failure

Exit the game

function game\_over\_guilty\_ending():

Display message indicating game over and reason for failure (guilty ending)

Exit the game

function game\_over\_bad\_ending():

Display message indicating game over and reason for failure (bad ending)

Exit the game

function game\_over\_worst\_ending():

Display message indicating game over and reason for failure (worst ending)

Exit the game

function game\_over\_best\_ending():

Display message indicating game over and reason for failure (best ending)

Exit the game

function game\_over\_best\_ending\_mystery\_solved():

Display message indicating successful completion of the game with best ending

Exit the game

function win\_game():

Display message indicating winning the game with the best ending

Exit the game

function game\_over\_lost\_ending():

Display message indicating game over and reason for failure (lost ending)

Exit the game

# Begin the game

function start\_game():

Display welcome message and introduction to the game

Call intro\_scenario() to start the first scenario

function intro\_scenario():

Display initial scenario and options for the player

Accept user input for the chosen action

If input is "1":

Call outcome\_a1()

Else if input is "2":

Call outcome\_b1()

Else:

Display error message and prompt user to try again

Recursively call intro\_scenario()