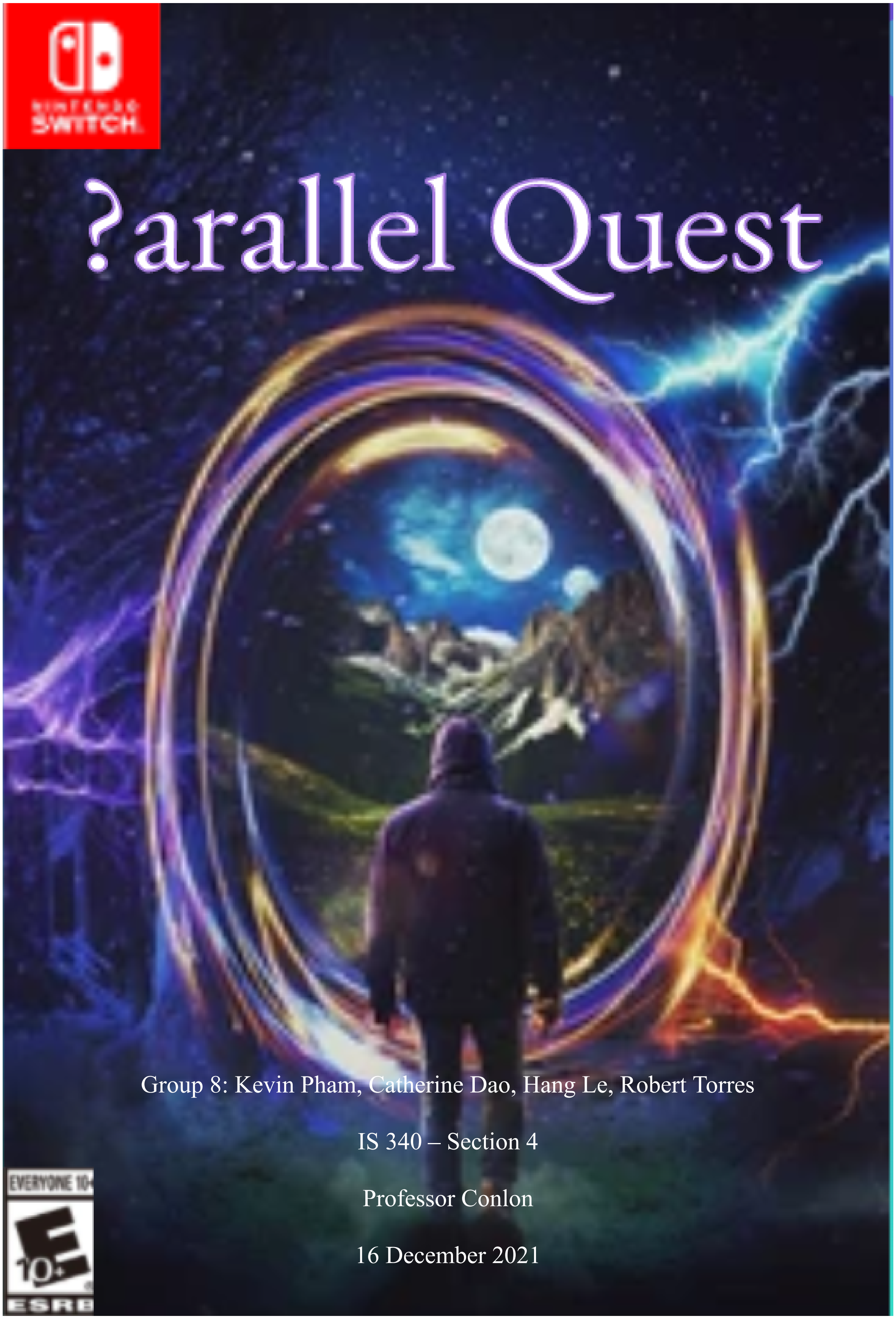


Group 8: Kevin Pham, Catherine Dao, Hang Le, Robert Torres

IS 340 – Section 4

Professor Conlon

16 December 2021

Introduction

In our game ?arallel Quest, the main character (you, the player) just turned 18 years old and had been given a mysterious box. Inside was a letter from your dad who has disappeared years prior due to an unknown expedition. The letter had stated that it was time for you to follow in his footsteps and to explore the mysterious Parallel world.

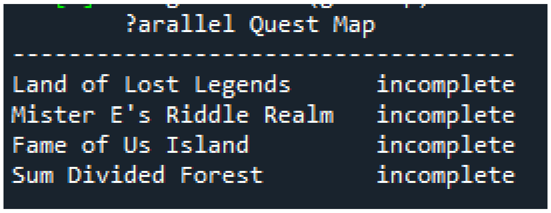
Also inside the box are 3 candles that give the user, upon ignition, access to the parallel world. There, guided by Mr. Elfo (your dad's old friend), you must conquer each land on the map to discover the true whereabouts of your father. After each completed land, the game will restock your candles to 3 if you have less than 3 candles.

Within the game, there are 4 main lands recorded on the progress map and two additional minigames. The first land you teleport to is the Land of Lost Legends where you meet Weorge Gashington, who tests your knowledge of American History in a game of trivia. Just as you are about to transport to the next point on the map, Mr. Elfo informs you of a delay in the teleportation portal, so you play Tic-Tac-Tea in the meantime. Once it was repaired, you continue to Mister E's Riddle Realm, where you must solve a series of riddles set by your father's old friend, Mister E and obtain his diary of adventures with your father. Then, you proceed to Fame of Us Island, where you participate in a popular game show with Steve Harvey as your host and also play the second minigame, WhoMan. If you win, you will receive a surprise birthday gift from your father. For the last land, Sum Divided Forest, you meet Mr. Pi. He presents several of his math-related question masterpieces, ranging from calculation to number guessing games. Only when you have completed the map will you be able to unlock the box your father left behind with an invitation letter to join your father on his journey.

Methods

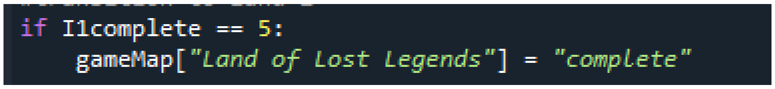
Foundation/Introduction

Before the game, we had to create a foundation. This included initializing the map, amount of lives, which are symbolized as candles, Click function counter completion per Island.



```
        ?arallel Quest Map
------------------------------------
Land of Lost Legends      incomplete
Mister E's Riddle Realm   incomplete
Fame of Us Island         incomplete
Sum Divided Forest        incomplete
```

For the map, we used Dictionaries in Chapter 8 from the example of keys and values. This is easier because, in the lecture, there was an example that gave us the freedom of creating a table and even updating the key's value. Also, we figured that a map would be a great way to keep track of the journey and something to refer back to. So this game was more focused on completing the map instead of gathering points. These were set in functions so we could refer back to them throughout the game and update after every island. Lives in our game are represented by candles and upon every island, you will restock your candle to 3 (if you complete the island with less than 3 candles). There are opportunities for the player to gain additional lives so the restocking code would only run if you advance with less than 3 candles. If you lose all 3 candles, the game will end immediately. Another function that was made was the Click function which allowed the narrative and dialogue to run at a pace instead of being thrown all at once into the console. It would display a certain scene, then ask you to click enter, then proceeds to the next part. Lastly, the counter for each island is to show if you completed the question on each island. It starts off at 0 and it will add one throughout the completion.



```
if I1complete == 5:
    gameMap["Land of Lost Legends"] = "complete"
```

This picture shows that if you completed the 5 questions from the first island (I1), then the map will update to show that you have completed it.

The Introduction to the game will show the Title of the game, along with a small narrative of the storyline which prompts the player to experience it. When reading Dad's letter, we used Ch. 7 and created and read a file, and then displayed it onto the console per line. Lastly, we allowed the user to input their name. If they leave it blank, they will be asked to re-enter. We kept it open to add numbers or make it a number as a name because in online games people will add numbers to their usernames as well.

Level 1: Land of Lost Legends

The first level is Land of Lost Legends, which is a multiple choice game on American history. The player will be introduced to the keeper of the land, who is Weorge Gashington. He will guide the player through 5 multiple choice questions. He doesn't give hints, but he gives compliments to encourage the player to keep going.

I made classes with methods to implement the questions. The first class is called Question. It has a constructor that initializes inputs for the question and its correct answer. There are 4 other methods within the class Question that allow editing the questions, editing the correct answers, displaying the questions, and checking the player's answer. The second class is MC, which is the class for multiple choice questions and is the subclass for class Question. It calls the constructor from the superclass that initializes inputs for the question and its correct answer. MC's constructor also initializes a list for choices of answer. Methods in the class MC include addChoice, in which we input a choice and that input will be added in a list of choices, and display, which prints out the question and the list of its choices for answer. After the transition to the land, after the player pressed Enter to continue, the first question pops up and prompts for the player's input for answer. There are 5 questions in this level. If the player inputs correct answer, it shows in the console that it is correct and the next question pops up. If the player inputs wrong answer, it shows in the console that it's incorrect, 1 candle is broken, and the question reappears for player to try again. If all 3 candles are broken, it says "Game Over" and the game stops there.

This level uses class and subclass from chapter 9 and 10 with functions from chapter 5 and list operation from chapter 6, the if/else statement from chapter 3, the for loop and string processing from chapter 4, and input string and format string from chapter 2. For improvement on this level, I would add input validation to make sure the player doesn't leave a blank input.

Level 2: Tic-Tac-Tea

For this second level, Tic-Tac-Tea is a game of tic-tac-toe displayed on a 3x3 grid with the opponent's moves being randomized. This game serves as a transition point to Mister E's Riddle Realm as it foreshadows the growing difficulty in the game by rewarding additional candles. It also helps explain that after each completed land, the player will return to portal space where Mr. Elfo is waiting to guide the player throughout their quest. (see code/ output in Appendix)

To create the game grid board, I made a dictionary from Ch.8 to store the X and O marks as value for a given key number that represents the position on the board. To keep track of the game, I created three global variables: turn (switches from X and O), count (keeps track of number of filled spaces), and win (0 = lose, 1 = win). For the logic behind the game, all of it is under a while loop from Ch. 4 that will continue to run while win is still 0 and count is not equal to 9 (boxes total on a 3x3 grid board). Within the while loop, many if statements from Ch 3 are used. The first if statement is used to indicate whose turn it is to either randomize the move or input the move location number. There is a long nested if statement that states that after 5 moves where a player has made at least 3 moves; if or else if the 3 row locations are filled with X or O, then print out the winning message. This game also utilizes exception handling from Ch. 7 to deal with incorrect types of inputs based on a list of all location # in Ch. 6.

As for suggestions that we received, we would have considered adding an algorithm where the random player will actually make better guesses the closer you are to winning. Difficulty with this game was getting the randomizer to play like a 2nd player.

Level 3: Mister E’s Riddle Realm

In our third level dealing with riddles to find Mister E (an old friend of your father), the player will be asked five riddles that determine the story path throughout the journey on this level. These riddle questions are all open-ended questions, so there are additional “hint” keywords to get a clue how many characters are expected in the answer. (see code/ output in Appendix)

On this level, the game uses mostly for loops, if/elif/else statements, and taking inputs from Ch.4, Ch. 3, and Ch.2. All five questions each had a for loop that indicates as long as the player still has candles, the question will be printed and prompt the player to enter an input. The elif statement that takes in the keywork “hint”, and else statement for all incorrect answers results in a reduction of candle “lives”. The riddles have answers that may be letters or numbers, but will not be case sensitive. The candle “lives” is defined externally as a global variable and printed story dialogue (Ch.1) one at a time using a Click() function defined at the beginning of the code (Ch. 5).

There were a lot of things we could have added to this level. For instance, we could add adaptive gameplay where there are multiple paths to reach the top with different riddles when the player has an extra option to try a different riddle or by letting the player pick left or right on their path, which will output different riddle questions. This means on each of the five riddle stages, there are multiple possible riddles and storyline to fit with those riddles. Another thing that could improve this level and storyline on this level is to scatter bits of information about Mister E and your father. With multiple possible paths the player can continue on, we could also create different endings: one where Mister E remains a mystery or another where the player interacts with Mister E.

Level 4: Fame of Us Island

Fame of Us Island acts as a transition from Mister E's Riddle Realm to the WhoMan mini game. In Fame of Us Island, the player is nominated to attend a game show hosted by the celebrity game show host Steve Harvey. Fame of Us Island consists of five multiple choice questions that are pop-culture related. The player loses a candle if they incorrectly answer any of the questions. To make this level more interesting, we added simulated Steve Harvey facial expressions. This was needed to promote realism since Steve Harvey is known in real life for his comical expressions when receiving answers. Our group chose to add Steve Harvey as the host because he is a well known pop-culture icon among younger audiences and our game is marketed towards a young demographic. In order to make the game seem more interesting, the questions in this level are more comical than those of our other levels. This level has extensive use of techniques taught in chapter 10 to implement multiple choice questions to the user.

Level 5: WhoMan

As a mini game after the Fame of Us game show, WhoMan follows the same rules as Hangman, but you get a hint that it is a name of a famous person and how many letters are in their first and last name. There are a limit of 7 attempts to complete the hangman for each failure. (see code/ output in Appendix)

To accomplish this, we started by creating a function with if and elif statements that will print different stages of the hangman based on numbers of turns (attempts) left (Ch. 5). There multiple global variables defined as word (the answer), guesses (the whole collection of guesses player inputted), guessSet (a set of all letters users has tried without duplicates), currentfailed (keeps track of empty spaces from the previous attempt), and currentSetlen (keeps track of the set length from previous attempt). The code runs in a while loop when players still have attempts to guess, and a for loop will check every letter in the word's index position (Ch.4). Three local variables are failed (number of empty blanks per attempt), guess (the new input per attempt), and item (a correct letter in word). An if/elif/else statement is used to determine if turns (attempts) are reduced for wrong guess or not for correct guess or repeated guess (Ch.3).

During the process, we ran into some difficulty when trying to debug when errors or undesirable outputs are displayed because of the loops, variables, or when considering different possible outcomes from the player's input. To improve the game, we carefully considered the feedback from our professor and classmates' feedback by adding a new variable called outputline to create a string where the output of correct/incorrect guesses are displayed horizontally, so now it displays vertically and horizontally.

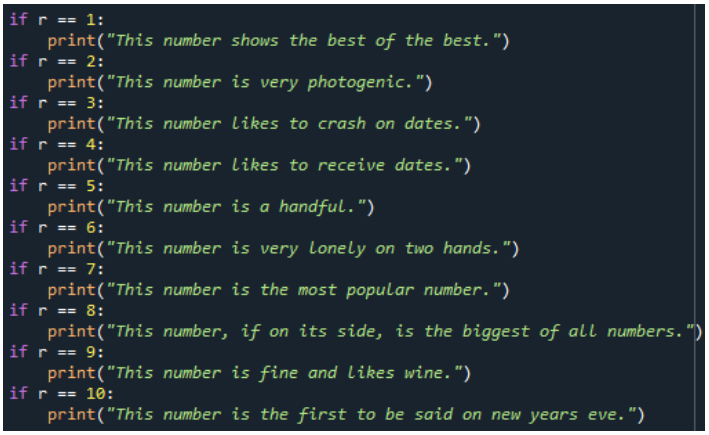
Level 6: Sum Divided Forest

This island is different compared to the previous ones because this focuses more on numbers and math application. The first question is a very basic addition problem where you are given a list (Chapter 6) of the height of the island keeper's family and you must find the total height in inches. We kept it in inches because, compared to feet, it would be in whole numbers. When asking for the answer, you have to put in an integer. If not, then there will be an exception that runs until an integer is submitted. The second question is the second part where it asks the user to find the average height of the family in inches while rounding to the nearest whole number. We kept it as a whole number again for the same reason as stated above. The rewards of these levels are additional lives. However, unlike one of the minigames, if they fail to answer, they will lose a life. This was added to give the player the flexibility of having more chances for the final question.

A problem that we constantly had in this question was how we would keep getting an error when inputting anything but an integer when "breaking" the game. A solution at the time was to keep the input and the answer set as a string instead of an integer. However, this would be another problem since we wanted to think more logically to give the users a chance to input a valid answer. This is because of our Click function utilizing the "enter" key so it was very easy for the player to accidentally submit a blank input as an answer. Therefore, we resorted to using the try and except to ensure that the inputs will be integers. If not, they will be asked to re-enter one without a loss of a candle.

Something that we wished to have improved on the second question was to implement an automatic round function to the player's input because the answer is set to a whole number (per instructions). So even if the player inputs the correct mathematical value, with decimals, it would be wrong according to the code. In addition, we wanted to improve how the list of heights should be different in every gameplay. We wanted to implement a randomized function that will generate a list of the family's heights from a range from 50 to 80 to keep the heights consistent. Therefore, this would allow the players, if replaying, to have a fresh set of heights to work with.

The final question is essentially a guessing game. A random number between 1 to 10 will be generated and the hints given will be automatically adjusted accordingly. We kept the hints with the purpose of narrowing the pool of numbers. To keep track of the hints, we added a new count variable to keep count when a new attempt is executed. This is to differentiate between the lives from attempts because there were errors when we gave hints when they had only 2 or 1 life left. But because of the additional opportunities of earning additional lives, a new approach was needed. So after each attempt, the count would add 1 to itself. Therefore, the second hint would run when the count equals 1, and the third hint would run if the count was 2. This question is also more "intensive" since the first part of this question is based on luck. The first hint, that is given automatically, is whether the number is odd or even. This gives the user a 25% chance, meaning that there is a 75% chance of losing a life. Then, the second hint shows if it is on the upper, lower, or neither of the spectrum. If it is on the upper or lower side, then the pool shortens down to half, but the possibility of losing another candle is still high. The only exception is if the random number was 5, which is in the middle(neither). Then, for the last hint, we wanted to implement a small riddle per number to make it more interesting.



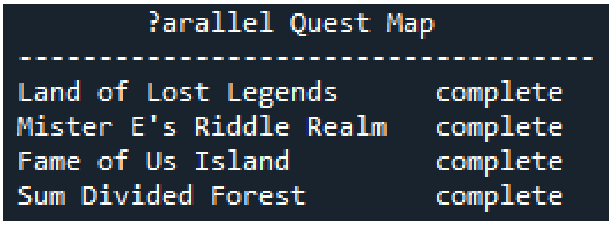
```
if r == 1:
    print("This number shows the best of the best.")
if r == 2:
    print("This number is very photogenic.")
if r == 3:
    print("This number likes to crash on dates.")
if r == 4:
    print("This number likes to receive dates.")
if r == 5:
    print("This number is a handful.")
if r == 6:
    print("This number is very lonely on two hands.")
if r == 7:
    print("This number is the most popular number.")
if r == 8:
    print("This number, if on its side, is the biggest of all numbers.")
if r == 9:
    print("This number is fine and likes wine.")
if r == 10:
    print("This number is the first to be said on new years eve.")
```

These are the unique hints and we made them based on social culture/slang, statistics and more. For example, 2 is very photogenic because a lot of people like to throw up a “peace” sign or even an aloha sign which uses up 2 fingers. 3 likes to crash on dates because socially, we know that 3 is a crowd.

One thing we considered was how difficult this question would be. We wanted to keep in mind that the player would have the maximum number of default candles(3) going into this question. So we structured this question around it. This is due to how some islands with a default of 3 candles, were able to complete it without losing a candle. In addition, it is more likely that the player will lose a good amount of candles on this question alone. So we wanted to keep it less difficult for the players with fewer candles going in, despite the previous questions rewarding additional candles.

Some things we wanted to add to this island are additional levels that either extend the first two questions or add an entirely new question. A possible extension to the first two questions could be to find the maximum and minimum height of the family. A question that we wanted to add but did not have time for was from the volume example from Chapter 5. An idea from our notes was that a box had been broken and the keeper gives you the volume of the box and you must find what are the dimensions needed to remake the box with the length, width, and height. This also seemed very difficult because of how many endless possibilities there are that would equal the given volume. For example, if the given volume was 64, then there would be a lot of combinations when asking the player to input the height, length, and width since it can be in any order and many numbers can fit in there. This was far from our current capacity.

End of game:



```
        ?arallel Quest Map
------------------------------------
Land of Lost Legends      complete
Mister E's Riddle Realm   complete
Fame of Us Island         complete
Sum Divided Forest        complete
```

This is the conclusion of our game once all of the islands are completed. The map should say all “complete” all around and then we wanted to keep this game open ended with the intentions of it being a first part of many games by creating another file of the father’s letter inviting the player to join him on the next quest where they will reunite.

Conclusion

The progression of the development of ?arallel Quest was very efficient but also very explorative. Most of the time, we would get on top of things but then new ideas would come up that would make the game much more interesting. These may include new games, different themes, certain twists, and more. In the beginning, we planned to have it as multiple choice games but then we had to rethink and find new ways to “answer” questions. For example, the riddle questions ask for word(s) as inputs, or tic tac tea asks you to select a grid, or the math problems make you use numbers. These were a few of the major discussions that were brought up in our group which built our team chemistry since we understood and trusted each other throughout the process. The evolution and thought process of our game has been very fun and creative. It all began with one idea, which was a journey through 4 islands, where each of the group members had the freedom to do what they wanted creatively. That was the biggest comment from the professor, which was how he saw how much fun we had with this project, which made the idea of learning how to code, more like a fun hobby. Therefore, it gave us a glimpse of computer science’s capability of turning a general idea into a functional program for entertainment. It also shows us how many programs we use today are through group efforts, like this project, and demonstrates how teamwork and creativity in the computer science world are very essential.

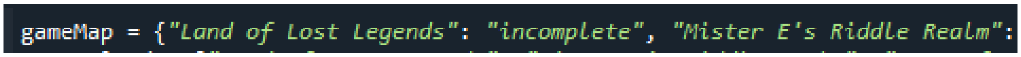
Appendix

<u>Image</u>:

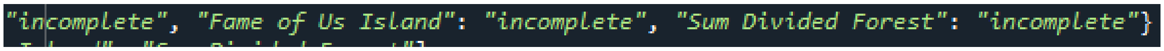
Foundation/Introduction

FIA(Map incompleted)

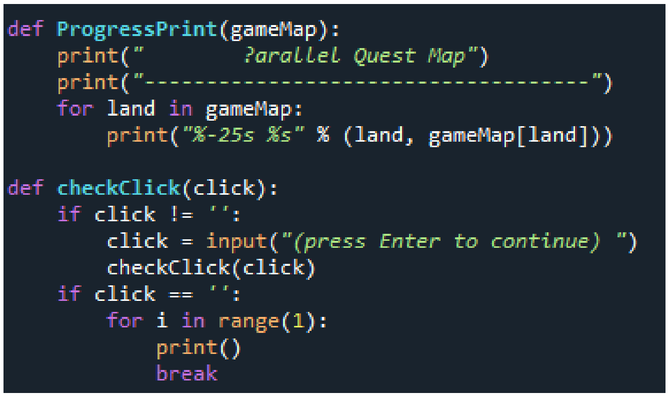
- **Input**

    - 

      ```
      gameMap = {"Land of Lost Legends": "incomplete", "Mister E's Riddle Realm":
      ```

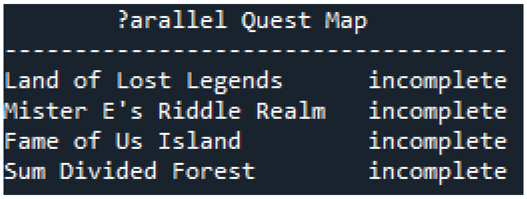
    - 

      ```
      "incomplete", "Fame of Us Island": "incomplete", "Sum Divided Forest": "incomplete"}
      ```

    - 

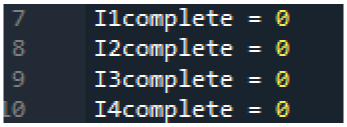
      ```
      def ProgressPrint(gameMap):
          print("        ?arallel Quest Map")
          print("------------------------------------")
          for land in gameMap:
              print("%-25s %s" % (land, gameMap[land]))

      def checkClick(click):
          if click != '':
              click = input("(press Enter to continue) ")
              checkClick(click)
          if click == '':
              for i in range(1):
                  print()
                  break
      ```

- **Output (console)**

    - 

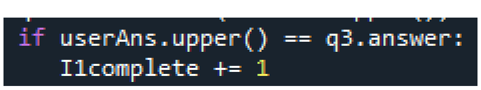
      ```
              ?arallel Quest Map
      ------------------------------------
      Land of Lost Legends      incomplete
      Mister E's Riddle Realm   incomplete
      Fame of Us Island         incomplete
      Sum Divided Forest        incomplete
      ```

FIB Island Initializing

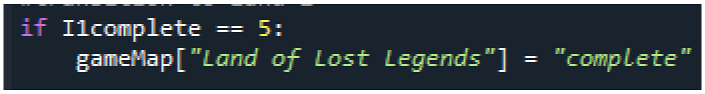
- 

  ```
  I1complete = 0
  I2complete = 0
  I3complete = 0
  I4complete = 0
  ```

- **Upon completion**

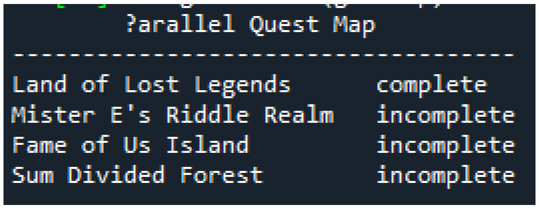


```
if userAns.upper() == q3.answer:
    I1complete += 1
```



```
if I1complete == 5:
    gameMap["Land of Lost Legends"] = "complete"
```

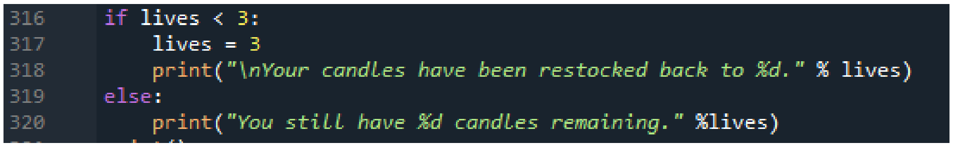
- **Output**



```
        ?arallel Quest Map
-----------------------------------
Land of Lost Legends       complete
Mister E's Riddle Realm    incomplete
Fame of Us Island          incomplete
Sum Divided Forest         incomplete
```

-

FIC (restock candle)

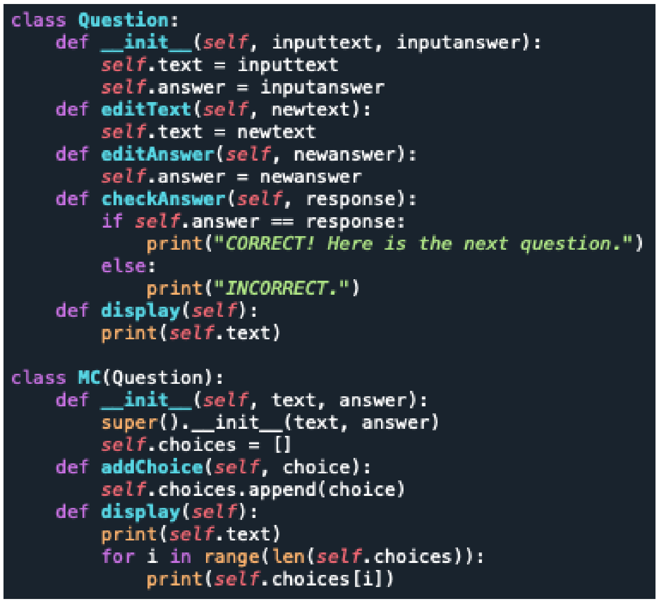


```
if lives < 3:
    lives = 3
    print("\nYour candles have been restocked back to %d." % lives)
else:
    print("You still have %d candles remaining." %lives)
```

-

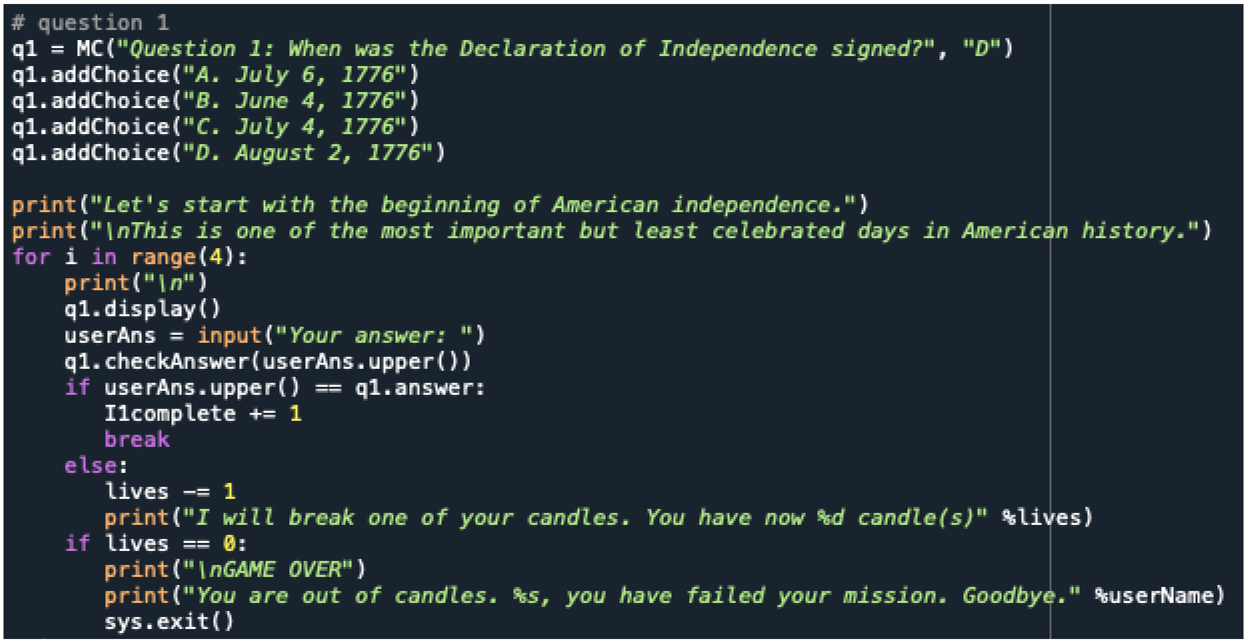
Land of Lost Legends

The class and subclass:



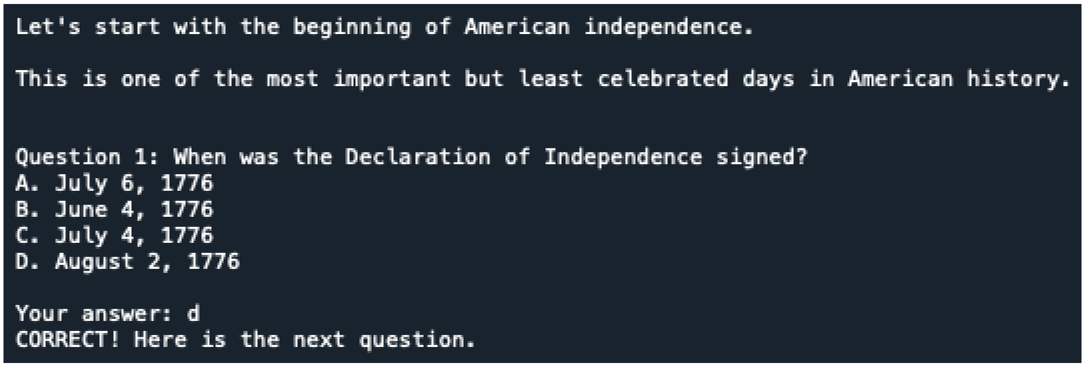
```
class Question:
    def __init__(self, inputtext, inputanswer):
        self.text = inputtext
        self.answer = inputanswer
    def editText(self, newtext):
        self.text = newtext
    def editAnswer(self, newanswer):
        self.answer = newanswer
    def checkAnswer(self, response):
        if self.answer == response:
            print("CORRECT! Here is the next question.")
        else:
            print("INCORRECT.")
    def display(self):
        print(self.text)

class MC(Question):
    def __init__(self, text, answer):
        super().__init__(text, answer)
        self.choices = []
    def addChoice(self, choice):
        self.choices.append(choice)
    def display(self):
        print(self.text)
        for i in range(len(self.choices)):
            print(self.choices[i])
```

The questions - Input:

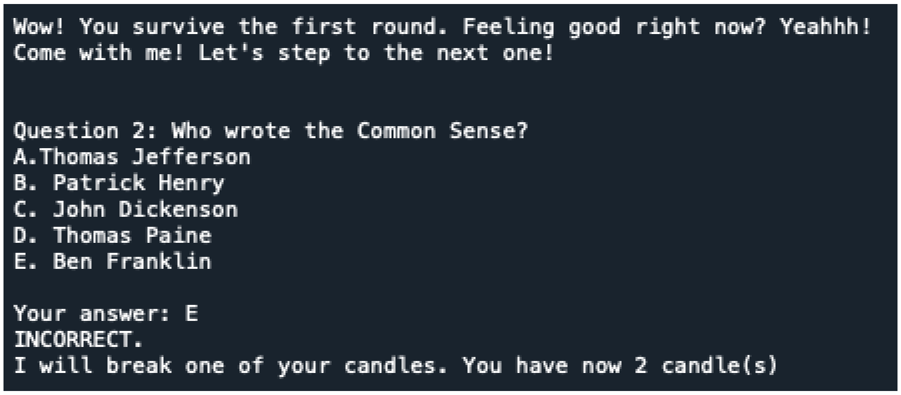


```
# question 1
q1 = MC("Question 1: When was the Declaration of Independence signed?", "D")
q1.addChoice("A. July 6, 1776")
q1.addChoice("B. June 4, 1776")
q1.addChoice("C. July 4, 1776")
q1.addChoice("D. August 2, 1776")

print("Let's start with the beginning of American independence.")
print("\nThis is one of the most important but least celebrated days in American history.")
for i in range(4):
    print("\n")
    q1.display()
    userAns = input("Your answer: ")
    q1.checkAnswer(userAns.upper())
    if userAns.upper() == q1.answer:
       I1complete += 1
       break
    else:
       lives -= 1
       print("I will break one of your candles. You have now %d candle(s)" %lives)
    if lives == 0:
       print("\nGAME OVER")
       print("You are out of candles. %s, you have failed your mission. Goodbye." %userName)
       sys.exit()
```

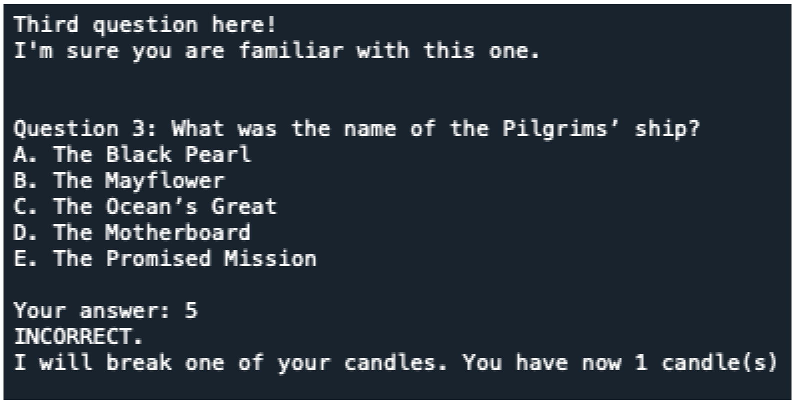
The questions - Output (with answer):



```
Let's start with the beginning of American independence.

This is one of the most important but least celebrated days in American history.


Question 1: When was the Declaration of Independence signed?
A. July 6, 1776
B. June 4, 1776
C. July 4, 1776
D. August 2, 1776

Your answer: d
CORRECT! Here is the next question.
```

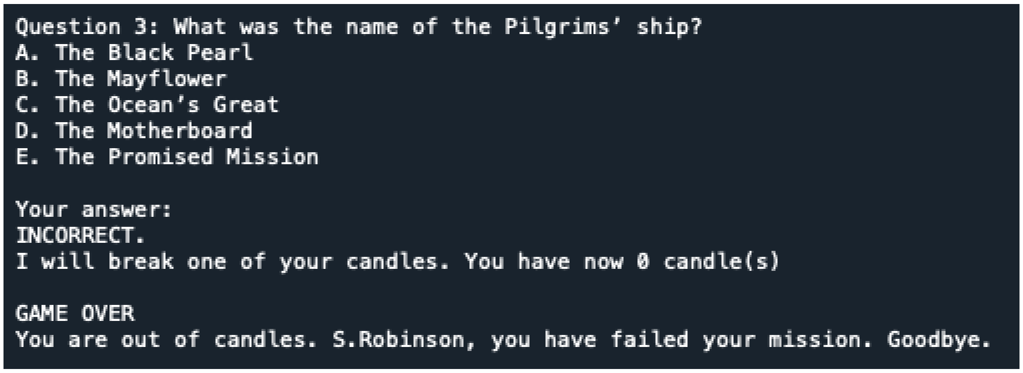


```
Wow! You survive the first round. Feeling good right now? Yeahhh!
Come with me! Let's step to the next one!


Question 2: Who wrote the Common Sense?
A.Thomas Jefferson
B. Patrick Henry
C. John Dickenson
D. Thomas Paine
E. Ben Franklin

Your answer: E
INCORRECT.
I will break one of your candles. You have now 2 candle(s)
```



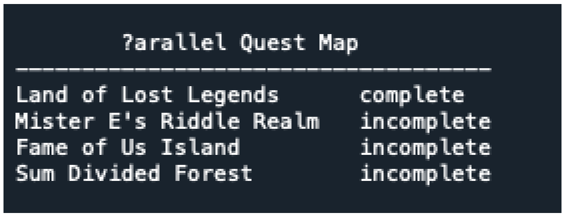
```
Third question here!
I'm sure you are familiar with this one.


Question 3: What was the name of the Pilgrims’ ship?
A. The Black Pearl
B. The Mayflower
C. The Ocean’s Great
D. The Motherboard
E. The Promised Mission

Your answer: 5
INCORRECT.
I will break one of your candles. You have now 1 candle(s)
```

(Blank input is counted as wrong answer. Once player fails 3 attempts, the game ends)



```
Question 3: What was the name of the Pilgrims’ ship?
A. The Black Pearl
B. The Mayflower
C. The Ocean’s Great
D. The Motherboard
E. The Promised Mission

Your answer:
INCORRECT.
I will break one of your candles. You have now 0 candle(s)

GAME OVER
You are out of candles. S.Robinson, you have failed your mission. Goodbye.
```

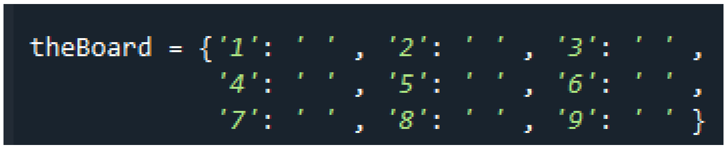
The map:



```
        ?arallel Quest Map
------------------------------------
Land of Lost Legends       complete
Mister E's Riddle Realm    incomplete
Fame of Us Island          incomplete
Sum Divided Forest         incomplete
```

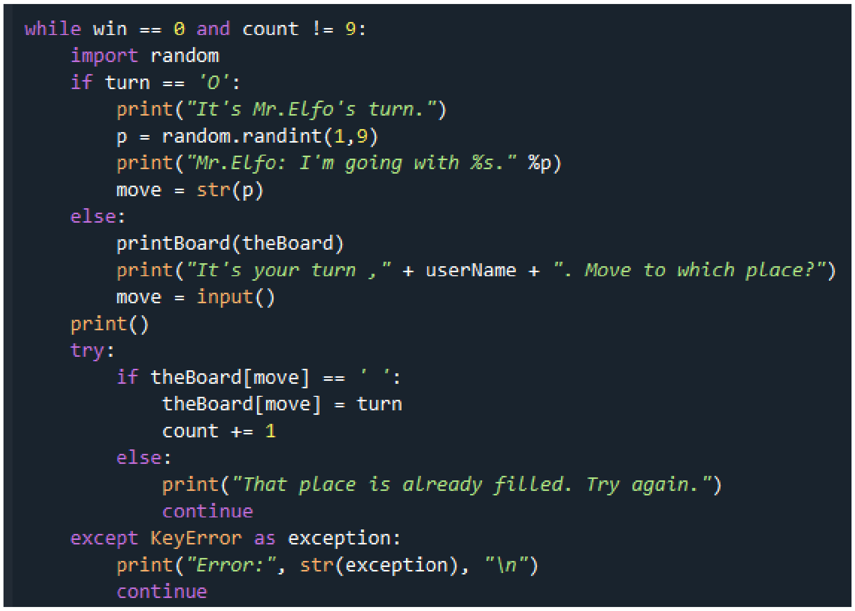
Tic-Tac-Tea

TTT1A (the grid board)



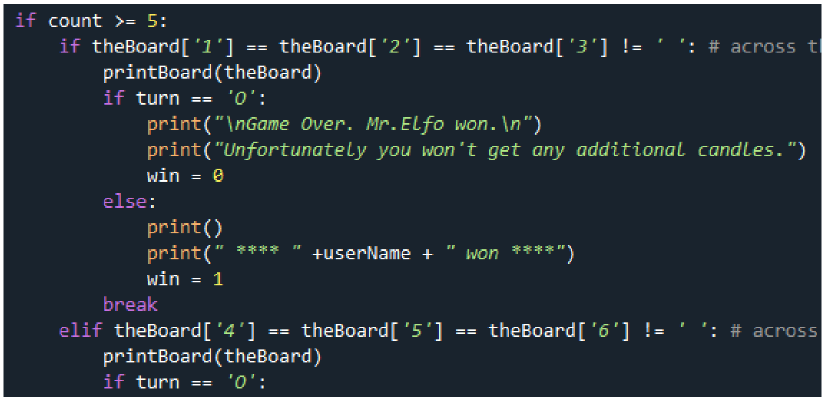
```
theBoard = {'1': ' ' , '2': ' ' , '3': ' ' ,
            '4': ' ' , '5': ' ' , '6': ' ' ,
            '7': ' ' , '8': ' ' , '9': ' ' }
```

TTT2A (while loop, if statement, exception handling)



```
while win == 0 and count != 9:
    import random
    if turn == 'O':
        print("It's Mr.Elfo's turn.")
        p = random.randint(1,9)
        print("Mr.Elfo: I'm going with %s." %p)
        move = str(p)
    else:
        printBoard(theBoard)
        print("It's your turn ," + userName + ". Move to which place?")
        move = input()
    print()
    try:
        if theBoard[move] == ' ':
            theBoard[move] = turn
            count += 1
        else:
            print("That place is already filled. Try again.")
            continue
    except KeyError as exception:
        print("Error:", str(exception), "\n")
        continue
```

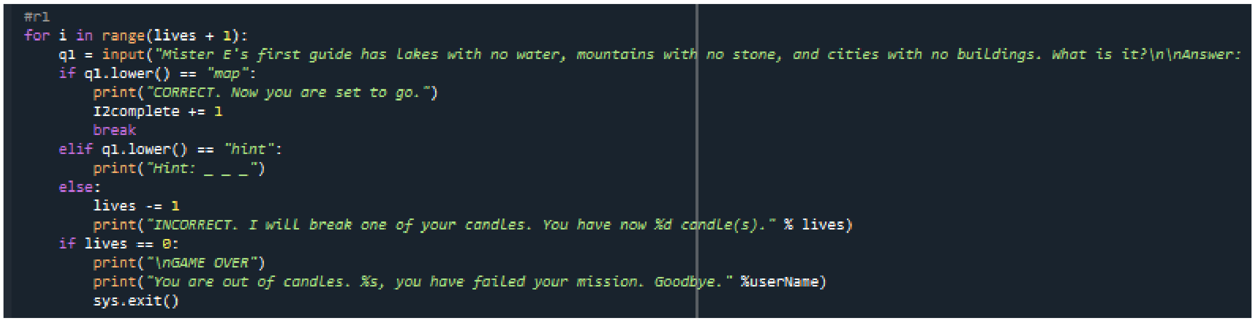
TTT3A (if statement determine result)



```
if count >= 5:
    if theBoard['1'] == theBoard['2'] == theBoard['3'] != ' ': # across t
        printBoard(theBoard)
        if turn == 'O':
            print("\nGame Over. Mr.Elfo won.\n")
            print("Unfortunately you won't get any additional candles.")
            win = 0
        else:
            print()
            print(" **** " +userName + " won ****")
            win = 1
        break
    elif theBoard['4'] == theBoard['5'] == theBoard['6'] != ' ': # across
        printBoard(theBoard)
        if turn == 'O':
```

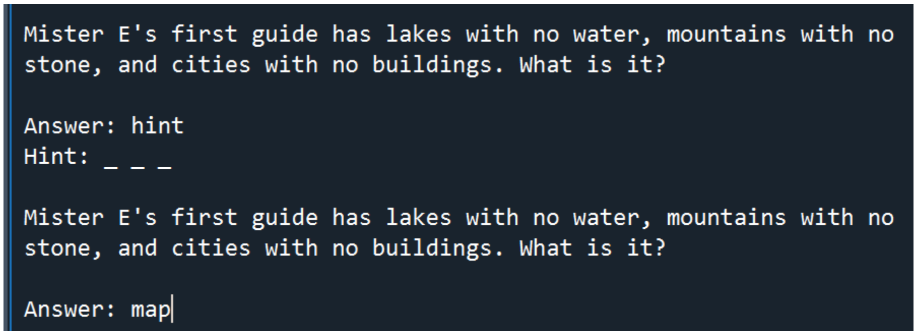
Mister E’s Riddle Realm

MERR1A (code)



```
#r1
for i in range(lives + 1):
    q1 = input("Mister E's first guide has lakes with no water, mountains with no stone, and cities with no buildings. What is it?\n\nAnswer:
    if q1.lower() == "map":
        print("CORRECT. Now you are set to go.")
        I2complete += 1
        break
    elif q1.lower() == "hint":
        print("Hint: _ _ _")
    else:
        lives -= 1
        print("INCORRECT. I will break one of your candles. You have now %d candle(s)." % lives)
    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye." %userName)
        sys.exit()
```

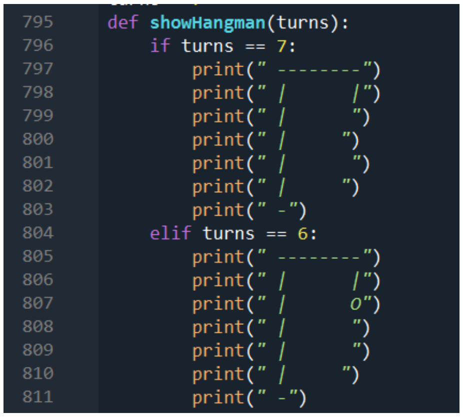
MERR2A (console output)



```
Mister E's first guide has lakes with no water, mountains with no
stone, and cities with no buildings. What is it?

Answer: hint
Hint: _ _ _

Mister E's first guide has lakes with no water, mountains with no
stone, and cities with no buildings. What is it?

Answer: map
```

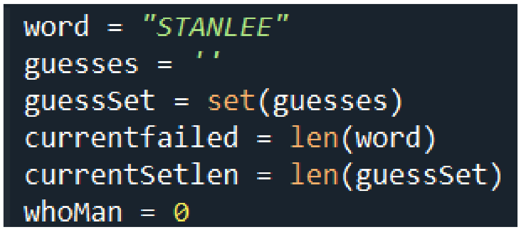
WhoMan

WM1A (function)



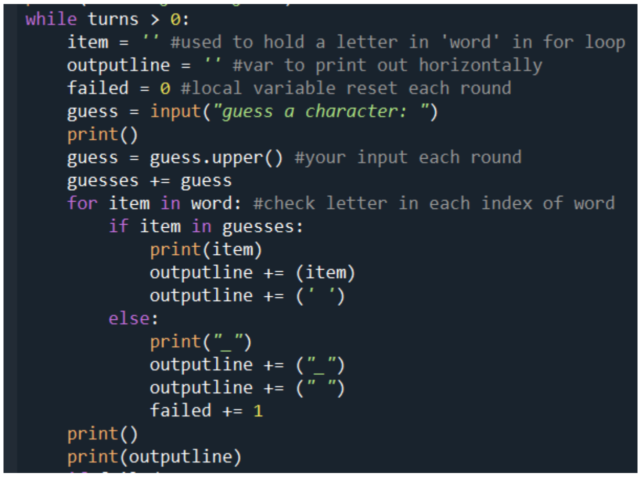
```
def showHangman(turns):
    if turns == 7:
        print(" --------")
        print(" |      |")
        print(" |      ")
        print(" |     ")
        print(" |      ")
        print(" |     ")
        print(" -")
    elif turns == 6:
        print(" --------")
        print(" |      |")
        print(" |      O")
        print(" |      ")
        print(" |      ")
        print(" |     ")
        print(" -")
```

WM2A (global variable)



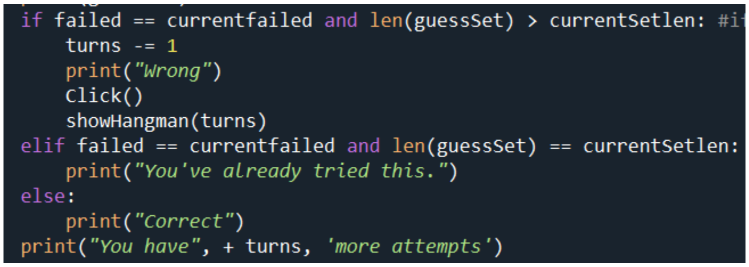
```
word = "STANLEE"
guesses = ''
guessSet = set(guesses)
currentfailed = len(word)
currentSetlen = len(guessSet)
whoMan = 0
```

WM3A (while/ for loop, local variable)



```
while turns > 0:
    item = '' #used to hold a letter in 'word' in for loop
    outputline = '' #var to print out horizontally
    failed = 0 #local variable reset each round
    guess = input("guess a character: ")
    print()
    guess = guess.upper() #your input each round
    guesses += guess
    for item in word: #check letter in each index of word
        if item in guesses:
            print(item)
            outputline += (item)
            outputline += (' ')
        else:
            print("_")
            outputline += ("_")
            outputline += (" ")
            failed += 1
    print()
    print(outputline)
```

WM4A (if statement track attempts)

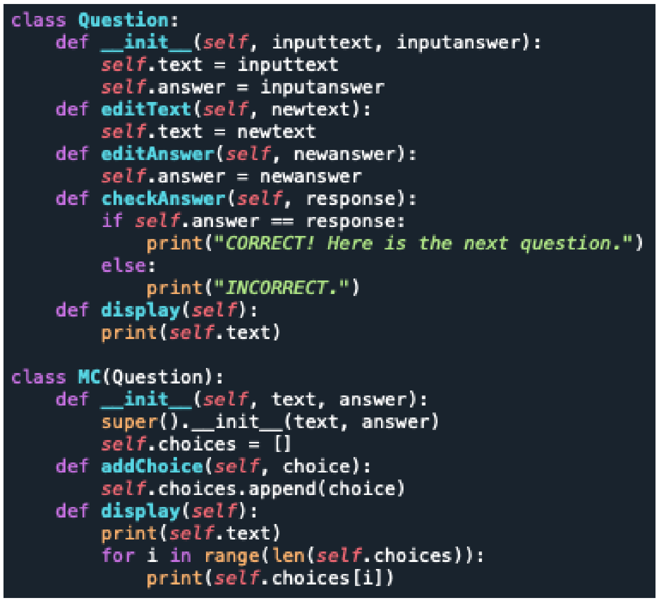


```
if failed == currentfailed and len(guessSet) > currentSetlen: #i
    turns -= 1
    print("Wrong")
    Click()
    showHangman(turns)
elif failed == currentfailed and len(guessSet) == currentSetlen:
    print("You've already tried this.")
else:
    print("Correct")
print("You have", + turns, 'more attempts')
```

Fame of Us Island

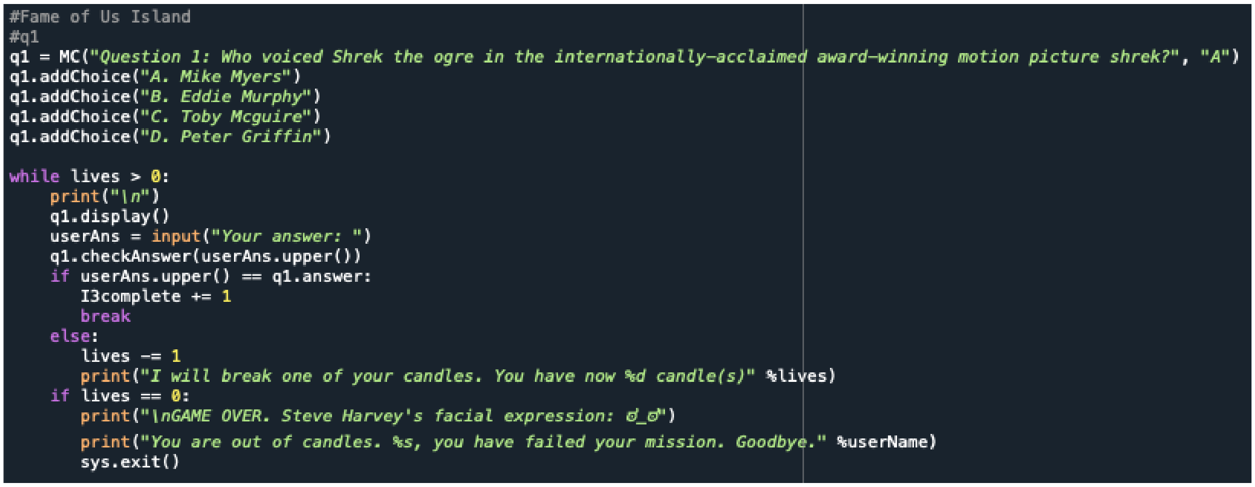
(similar codes and arrangements to Land of Lost Legends)

The class and subclass:



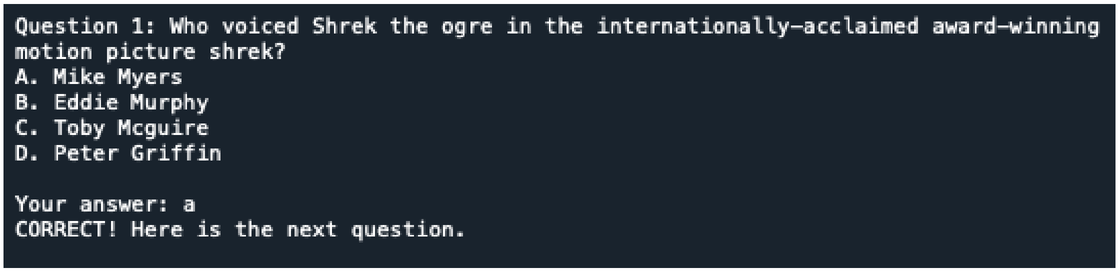
```
class Question:
    def __init__(self, inputtext, inputanswer):
        self.text = inputtext
        self.answer = inputanswer
    def editText(self, newtext):
        self.text = newtext
    def editAnswer(self, newanswer):
        self.answer = newanswer
    def checkAnswer(self, response):
        if self.answer == response:
            print("CORRECT! Here is the next question.")
        else:
            print("INCORRECT.")
    def display(self):
        print(self.text)

class MC(Question):
    def __init__(self, text, answer):
        super().__init__(text, answer)
        self.choices = []
    def addChoice(self, choice):
        self.choices.append(choice)
    def display(self):
        print(self.text)
        for i in range(len(self.choices)):
            print(self.choices[i])
```

The question - Input:

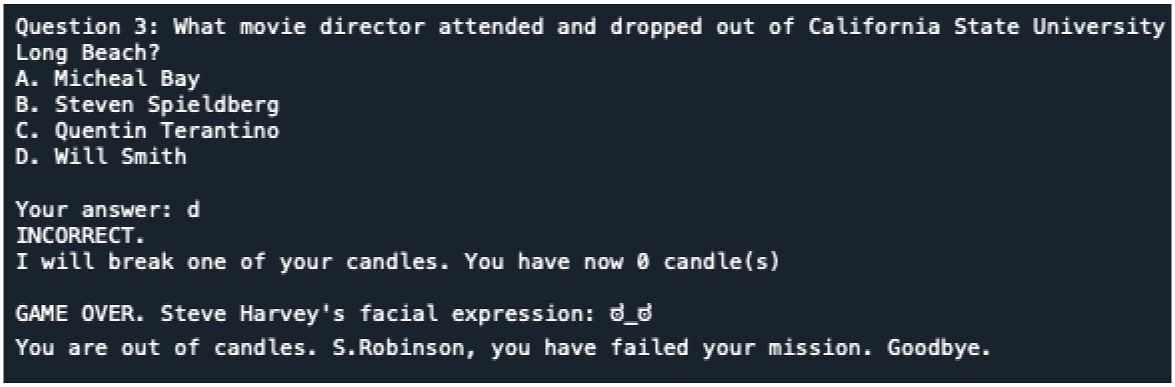


```
#Fame of Us Island
#q1
q1 = MC("Question 1: Who voiced Shrek the ogre in the internationally-acclaimed award-winning motion picture shrek?", "A")
q1.addChoice("A. Mike Myers")
q1.addChoice("B. Eddie Murphy")
q1.addChoice("C. Toby Mcguire")
q1.addChoice("D. Peter Griffin")

while lives > 0:
    print("\n")
    q1.display()
    userAns = input("Your answer: ")
    q1.checkAnswer(userAns.upper())
    if userAns.upper() == q1.answer:
       I3complete += 1
       break
    else:
       lives -= 1
       print("I will break one of your candles. You have now %d candle(s)" %lives)
    if lives == 0:
       print("\nGAME OVER. Steve Harvey's facial expression: ಠ_ಠ")
       print("You are out of candles. %s, you have failed your mission. Goodbye." %userName)
       sys.exit()
```

The question - Output (with answer):

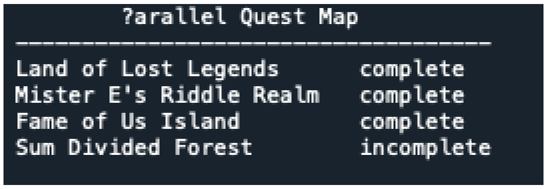


```
Question 1: Who voiced Shrek the ogre in the internationally-acclaimed award-winning
motion picture shrek?
A. Mike Myers
B. Eddie Murphy
C. Toby Mcguire
D. Peter Griffin

Your answer: a
CORRECT! Here is the next question.
```



```
Question 3: What movie director attended and dropped out of California State University
Long Beach?
A. Micheal Bay
B. Steven Spieldberg
C. Quentin Terantino
D. Will Smith

Your answer: d
INCORRECT.
I will break one of your candles. You have now 0 candle(s)

GAME OVER. Steve Harvey's facial expression: ಠ_ಠ
You are out of candles. S.Robinson, you have failed your mission. Goodbye.
```

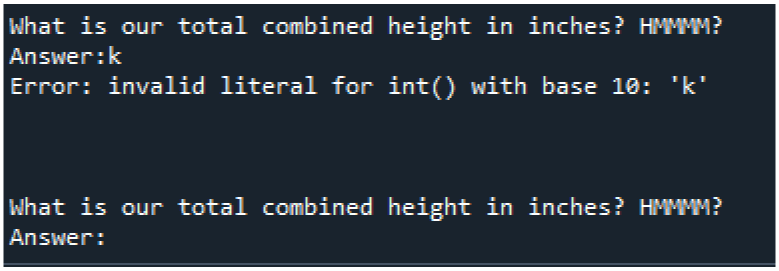
The map:



```
         ?arallel Quest Map
------------------------------------
Land of Lost Legends       complete
Mister E's Riddle Realm    complete
Fame of Us Island          complete
Sum Divided Forest         incomplete
```

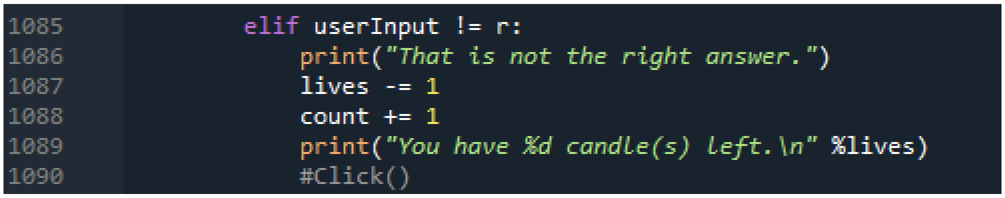
Sum Divided Forest

SDF Q1 (if input a non-integer)

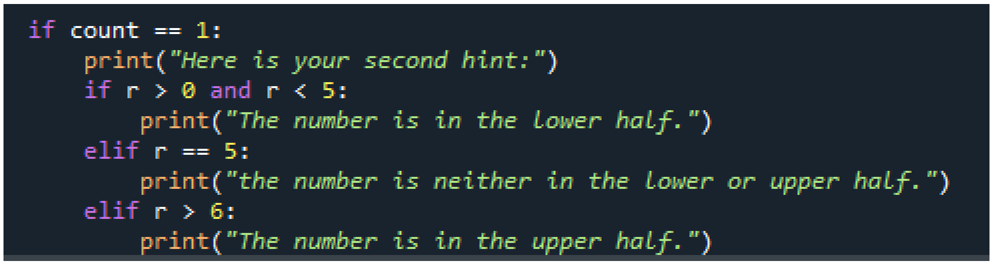
- 

```
What is our total combined height in inches? HMMMM?
Answer:k
Error: invalid literal for int() with base 10: 'k'



What is our total combined height in inches? HMMMM?
Answer:
```

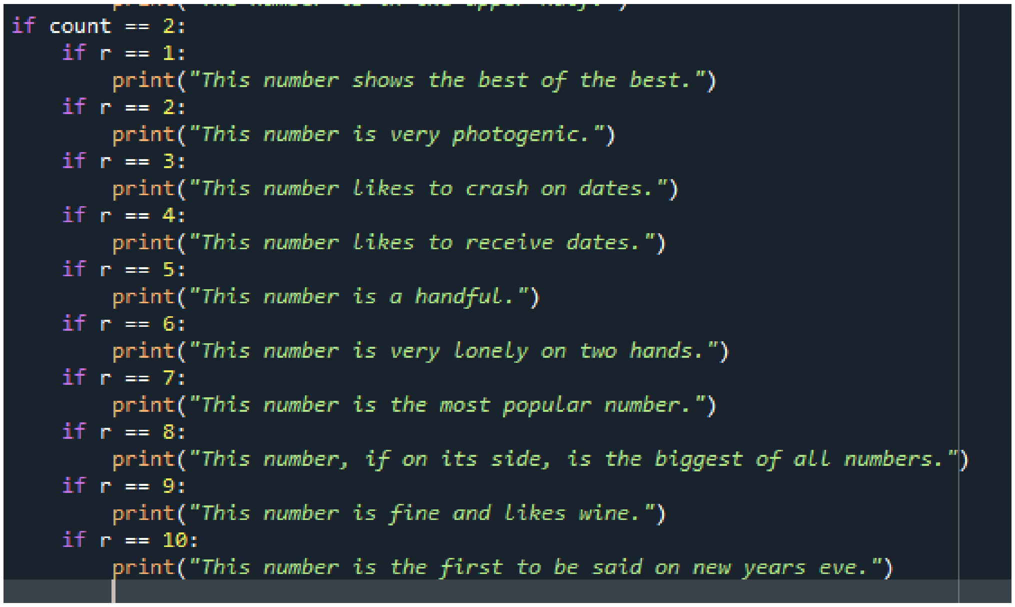
SDF Q3A: track of hints upon attempts with count



```
        elif userInput != r:
            print("That is not the right answer.")
            lives -= 1
            count += 1
            print("You have %d candle(s) left.\n" %lives)
            #Click()
```

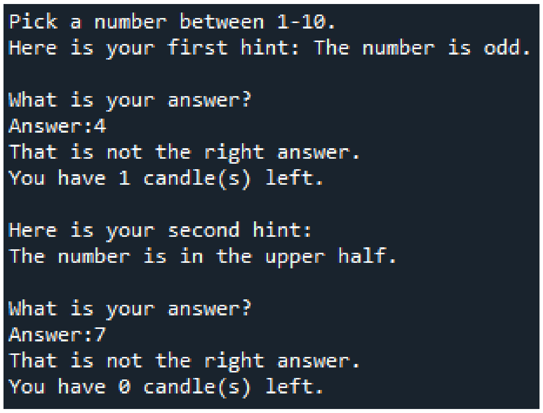


```
if count == 1:
    print("Here is your second hint:")
    if r > 0 and r < 5:
        print("The number is in the lower half.")
    elif r == 5:
        print("the number is neither in the lower or upper half.")
    elif r > 6:
        print("The number is in the upper half.")
```



```
        print("The number is in the upper half.")
if count == 2:
    if r == 1:
        print("This number shows the best of the best.")
    if r == 2:
        print("This number is very photogenic.")
    if r == 3:
        print("This number likes to crash on dates.")
    if r == 4:
        print("This number likes to receive dates.")
    if r == 5:
        print("This number is a handful.")
    if r == 6:
        print("This number is very lonely on two hands.")
    if r == 7:
        print("This number is the most popular number.")
    if r == 8:
        print("This number, if on its side, is the biggest of all numbers.")
    if r == 9:
        print("This number is fine and likes wine.")
    if r == 10:
        print("This number is the first to be said on new years eve.")
```

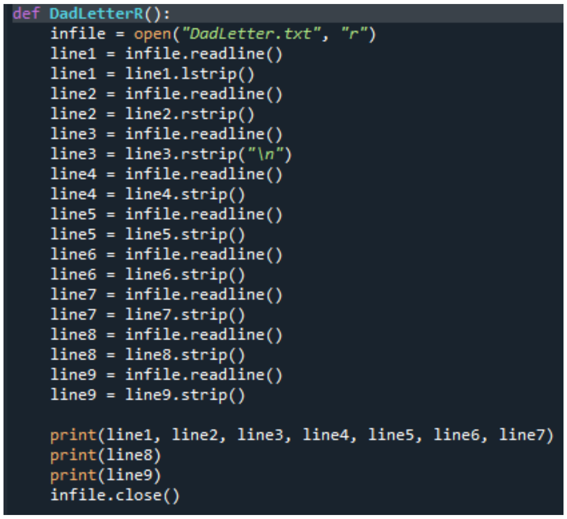
Output



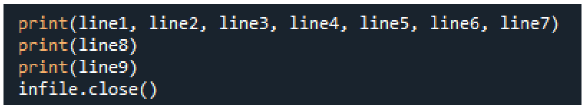
```
Pick a number between 1-10.
Here is your first hint: The number is odd.

What is your answer?
Answer:4
That is not the right answer.
You have 1 candle(s) left.

Here is your second hint:
The number is in the upper half.

What is your answer?
Answer:7
That is not the right answer.
You have 0 candle(s) left.
```

End of Game

EoGA (Dad letter

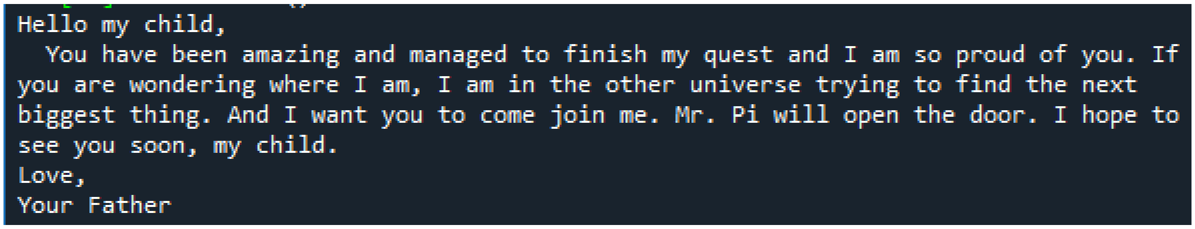


```
def DadLetterR():
    infile = open("DadLetter.txt", "r")
    line1 = infile.readline()
    line1 = line1.lstrip()
    line2 = infile.readline()
    line2 = line2.rstrip()
    line3 = infile.readline()
    line3 = line3.rstrip("\n")
    line4 = infile.readline()
    line4 = line4.strip()
    line5 = infile.readline()
    line5 = line5.strip()
    line6 = infile.readline()
    line6 = line6.strip()
    line7 = infile.readline()
    line7 = line7.strip()
    line8 = infile.readline()
    line8 = line8.strip()
    line9 = infile.readline()
    line9 = line9.strip()

    print(line1, line2, line3, line4, line5, line6, line7)
    print(line8)
    print(line9)
    infile.close()
```



```
print(line1, line2, line3, line4, line5, line6, line7)
print(line8)
print(line9)
infile.close()
```

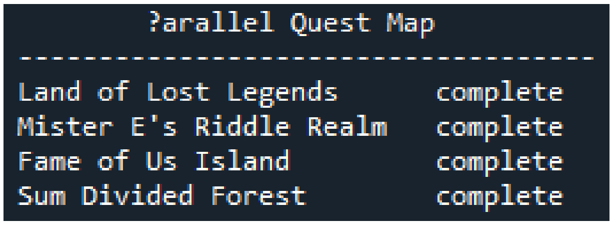
Output



```
Hello my child,
  You have been amazing and managed to finish my quest and I am so proud of you. If
you are wondering where I am, I am in the other universe trying to find the next
biggest thing. And I want you to come join me. Mr. Pi will open the door. I hope to
see you soon, my child.
Love,
Your Father
```

-

EOGB(completed map)



```
         ?arallel Quest Map
------------------------------------
Land of Lost Legends        complete
Mister E's Riddle Realm     complete
Fame of Us Island           complete
Sum Divided Forest          complete
```

Python Code:

```
gameMap = {"Land of Lost Legends": "incomplete", "Mister E's Riddle Realm":
"incomplete", "Fame of Us Island": "incomplete", "Sum Divided Forest": "incomplete"}
gameIsland = ["Land of Lost Legends", "Mister E's Riddle Realm", "Fame of Us Island",
"Sum Divided Forest"]
count = 0
lives = 3


I1complete = 0
I2complete = 0
I3complete = 0
I4complete = 0

import sys

def ProgressPrint(gameMap):
    print("        ?arallel Quest Map")
    print("-----------------------------------")
    for land in gameMap:
        print("%-25s %s" % (land, gameMap[land]))

def checkClick(click):
    if click != '':
        click = input("(press Enter to continue) ")
        checkClick(click)
    if click == '':
        for i in range(1):
            print()
            break


def Click():
    print()
    input("(press Enter to continue)")
    print("\n")


def DadLetterW():
    outfile = open("DadLetter.txt", "w")
    outfile.write("Hello my child,")
    outfile.write("\n")
    outfile.write("\nYou have been amazing and managed to finish my quest and I am so
proud of you.")
    outfile.write("\nIf you are wondering where I am, I am in the other universe trying to
find the")
    outfile.write("\nnext biggest thing. And I want you to come join me. Mr. Pi will open the
door.")
```

```
    outfile.write("\nI hope to see you soon, my child.\n")
    outfile.write("\n")
    outfile.write("Love,\n")
    outfile.write("Your Father")
    outfile.close()

def DadLetterR():
    infile = open("DadLetter.txt", "r")
    line1 = infile.readline()
    line1 = line1.lstrip()
    line2 = infile.readline()
    line2 = line2.rstrip()
    line3 = infile.readline()
    line3 = line3.rstrip("\n")
    line4 = infile.readline()
    line4 = line4.strip()
    line5 = infile.readline()
    line5 = line5.strip()
    line6 = infile.readline()
    line6 = line6.strip()
    line7 = infile.readline()
    line7 = line7.strip()
    line8 = infile.readline()
    line8 = line8.strip()
    line9 = infile.readline()
    line9 = line9.strip()

    print(line1, line2, line3, line4, line5, line6, line7)
    print(line8)
    print(line9)
    infile.close()

print("Before starting the game, please play in full screen (expanded console)")
Click()
print("")
print("
----------------------------------------------------------------------------------------------------------------
-------------------")
print("     /  \      ______                                          ________
/   | ")
print("    /   |     /     \                        |    |           |           /       \
/    /      ")
print("   /    |    |      |                        |    |           |          |        |                        |
/    /         ")
print("  /    /          /   ___        ___   ____      |    |          |          |          |
____      |       /    /           ")
```

```
print("  /    /         /    /    \|    | /    /    \|    |    |    _____   |         |          |     |    |  _____  /
___|___    /    /         ")
print(" /    /         |    |     ||   |/     |     ||    |    |  /____\  |         |       __  |    |     | /____\  \___
|      /    /        ")
print("/    /          .    \____/|_  |       \____/|_    |     |  \_____    |          \______/__\/___
\____/| \_____   ____\      |     /    /     ")
print("/  /
\   /     ")
print("---------------------------------------------------------------------------------------------------------------
------------------------ ")
print("\n \n \n \nGame Loading: 30%")
Click()
print("Game Loaded 100%")
#intro")
Click()
print("It is your 18th birthday and a box unexpectedly appeared in front of you. Inside
are three candles and a note. In that note reads...")
Click()
print("Happy birthday son. It is time for you to explore the path that I once did 3 years. I
am probably still exploring right now but hopefully, your journey will reunite us very
soon.")
print("Light up the candles, and close your eyes. When you open them again, your
journey will begin.")
```

```
print("Love,")
print("Dad")
Click()
print("You light up the candles... and close your eyes... ")
print("Your head gets swirled and hear sonic screams... Then you hear a voice...")
Click()
print("Hello? Hello? Are you alive human?")
Click()
print("You open your eyes and stand up...")
Click()
print("Ahh welcome. My name is Mr. Elfo. You must be the child of Dr. Robinson. We've been expecting you. What was your name?")
nameC = 0
while nameC == 0:
    userName = input("Name: ")
    if len(userName) >= 1:
        userName = userName
        break
    else:
        print("That is not a valid name, please re-enter a name.")
print("")
print("Ahh, of course, of course. How could I forget. Your father told me your name was %s. It is an honor to meet you. I suppose you are following in the same footsteps as
your father. I expect a lot from you since he is one of our famed conquerors."
%userName)
print("")
print("Before I let you go, please do pay very close attention to these rules.")
Click()

print("Rule #1: \n    You have those 3 candles. These represents your access to this
Parallel World. \n    Failing to answer a question correctly, will result to it breaking.")
print("Rule #2: \n    You must complete the 4 islands in order according to the map.")
print("Rule #3: \n    You should answer the questions with your own mind and
knowledge... Without any help.")
print("Rule #4: \n    If you break all of your candles, You lose access to this Parallel
World... Forever.")

Click()
print("\nWithout further ado, here is your map.\n")
ProgressPrint(gameMap)

Click()
print("\nYour father began his journey at the Land of Lost Legends. Therefore, you shall
too. There, you will meet the legendary Weorge Gashington. I shall guide you to your
next island upon completion. Good Luck!\n")
Click()
```

```
print("\n")
print("Teleported to %s..." % gameIsland[0])
print("\n")
Click()



#Land of Lost Legends
print()
print("\nWelcome %s to the Land of Lost Legends, my name is Weorge Gashington.
Yes, very similar but just as handsome as your first president!\n\nI hope you know the
history of your predecessors because you will be tested on them!" %userName)
print("\nDon't worry child. I will guide you through this wonderful land.\nUse the best of
your memory to break these American history challenges.")
print("\nOkay now, here is your first question.")
Click()

class Question:
    def __init__(self, inputtext, inputanswer):
        self.text = inputtext
        self.answer = inputanswer
    def editText(self, newtext):
        self.text = newtext
    def editAnswer(self, newanswer):
        self.answer = newanswer
    def checkAnswer(self, response):
        if self.answer == response:
            print("CORRECT! Here is the next question.")
        else:
            print("INCORRECT.")
    def display(self):
        print(self.text)


class MC(Question):
    def __init__(self, text, answer):
        super().__init__(text, answer)
        self.choices = []
    def addChoice(self, choice):
        self.choices.append(choice)
    def display(self):
        print(self.text)
        for i in range(len(self.choices)):
            print(self.choices[i])


# question 1
q1 = MC("Question 1: When was the Declaration of Independence signed?", "D")
q1.addChoice("A. July 6, 1776")
```

```
q1.addChoice("B. June 4, 1776")
q1.addChoice("C. July 4, 1776")
q1.addChoice("D. August 2, 1776")

print("Let's start with the beginning of American independence.")
print("\nThis is one of the most important but least celebrated days in American
history.")
for i in range(4):
  print("\n")
  q1.display()
  userAns = input("Your answer: ")
  q1.checkAnswer(userAns.upper())
  if userAns.upper() == q1.answer:
    l1complete += 1
    break
  else:
    lives -= 1
    print("I will break one of your candles. You have now %d candle(s)" %lives)
  if lives == 0:
    print("\nGAME OVER")
    print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
    sys.exit()
```

```
#q2
q2 = MC("Question 2: Who wrote the Common Sense?", "D")
q2.addChoice("A.Thomas Jefferson")
q2.addChoice("B. Patrick Henry")
q2.addChoice("C. John Dickenson")
q2.addChoice("D. Thomas Paine")
q2.addChoice("E. Ben Franklin")

print("\n")
print("Wow! You survive the first round. Feeling good right now? Yeahhh!")
print("Come with me! Let's step to the next one!")
for i in range(4):
  print("\n")
  q2.display()
  userAns = input("Your answer: ")
  q2.checkAnswer(userAns.upper())
  if userAns.upper() == q2.answer:
    l1complete += 1
    break
  else:
    lives -= 1
    print("I will break one of your candles. You have now %d candle(s)" %lives)
  if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()

#q3
q3 = MC("Question 3: What was the name of the Pilgrims’ ship?", "B")
q3.addChoice("A. The Black Pearl")
q3.addChoice("B. The Mayflower")
q3.addChoice("C. The Ocean’s Great")
q3.addChoice("D. The Motherboard")
q3.addChoice("E. The Promised Mission")

print("\n")
print("Third question here!\nI'm sure you are familiar with this one.")
for i in range(4):
    print("\n")
    q3.display()
    userAns = input("Your answer: ")
    q3.checkAnswer(userAns.upper())
    if userAns.upper() == q3.answer:
        l1complete += 1
        break
```

```
  else:
    lives -= 1
    print("I will break one of your candles. You have now %d candle(s)" %lives)
  if lives == 0:
    print("\nGAME OVER")
    print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
    sys.exit()
#q4
q4 = MC("Question 4: What City was the first capital of the United States?", "B")
q4.addChoice("A. Washington D.C.")
q4.addChoice("B. New York City")
q4.addChoice("C. Philadelphia")
q4.addChoice("D. Charleston")
q4.addChoice("E. Boston")

print("\n")
print("You are moving closer to the end of this land. Just keep up this pace!")
for i in range(4):
  print("\n")
  q4.display()
  userAns = input("Your answer: ")
  q4.checkAnswer(userAns.upper())
  if userAns.upper() == q4.answer:
    l1complete += 1
    break
  else:
    lives -= 1
    print("I will break one of your candles. You have now %d candle(s)" %lives)
  if lives == 0:
    print("\nGAME OVER")
    print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
    sys.exit()
#q5
q5 = MC("Question 5: Which US president did NOT hold 2 terms?", "E")
q5.addChoice("A. George W. Bush")
q5.addChoice("B. Jimmy Carter")
q5.addChoice("C. William Howard Taft")
q5.addChoice("D. A & C")
q5.addChoice("E. B & C")
q5.addChoice("F. None of the above")

print("\n")
print("Oh my my, last round already?")
print("You'd better watch out, because this one is going to be tricky *_*")
```

```
for i in range(4):
  print("\n")
  q5.display()
  userAns = input("Your answer: ")
  if userAns.upper() == q5.answer:
    print("You got it CORRECT!")
    l1complete += 1
    break
  else:
    lives -= 1
    print("INCORRECT. I will break one of your candles. You have now %d candle(s)"
%lives)
  if lives == 0:
    print("\nGAME OVER")
    print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
    sys.exit()


#transition to land 2
if l1complete == 5:
  gameMap["Land of Lost Legends"] = "complete"


print("\n")
print("Mr. Elfo reappears...")
Click()
click = input("Hello %s. Congratulations! You have conquered the Land of Lost
Legends.\nLook at your map, it has updated.\n\n(press Enter to continue)" %userName)
print("\n")
checkClick(click)
ProgressPrint(gameMap)
Click()
print()
print("The next stop on your journey is at %s." % gameIsland[1])
if lives < 3:
   lives = 3
   print("\nYour candles have been restocked back to %d." % lives)
else:
   print("You still have %d candles remaining." %lives)
print()
Click()
##########################################################################
########
#NEW ADDITION: Tic-Tac-Tea
print("Ooh, it seems like it will take a couple more minutes for the teleportation device to
be ready for you.")
print()
print("Let's play Tic-Tac-Tea, a classic game of tic-tac-toe during our teatime. As a bonus, I'll even add 2 more candles if you win.")
Click()


firstBoard = {'1': '1' , '2': '2' , '3': '3' ,
            '4': '4' , '5': '5' , '6': '6' ,
            '7': '7' , '8': '8' , '9': '9' }


theBoard = {'1': ' ' , '2': ' ' , '3': ' ' ,
            '4': ' ' , '5': ' ' , '6': ' ' ,
            '7': ' ' , '8': ' ' , '9': ' ' }
board_keys = []


for key in theBoard:
    board_keys.append(key)


def printBoard(board):
    print(board['1'] + '|' + board['2'] + '|' + board['3'])
    print('-+-+-')
    print(board['4'] + '|' + board['5'] + '|' + board['6'])
    print('-+-+-')
    print(board['7'] + '|' + board['8'] + '|' + board['9'])
print("Here's the table setup for this classic tic-tac-toe.")
print("Enter only an integer value between 1-9 corresponding to the spots on the
board.")
printBoard(firstBoard)
print()
Click()
print("Now let's begin!")
Click()
turn = 'X'
count = 0
win = 0

while win == 0 and count != 9:
    import random
    if turn == 'O':
        print("It's Mr.Elfo's turn.")
        p = random.randint(1,9)
        print("Mr.Elfo: I'm going with %s." %p)
        move = str(p)
    else:
        printBoard(theBoard)
        print("It's your turn ," + userName + ". Move to which place?")
        move = input()
    print()
    try:
        if theBoard[move] == ' ':
            theBoard[move] = turn
            count += 1
        else:
            print("That place is already filled. Try again.")
            continue
    except KeyError as exception:
        print("Error:", str(exception), "\n")
        continue

    if count >= 5:
        if theBoard['1'] == theBoard['2'] == theBoard['3'] != ' ': # across the top
            printBoard(theBoard)
            if turn == 'O':
                print("\nGame Over. Mr.Elfo won.\n")
                print("Unfortunately you won't get any additional candles.")
                win = 0
            else:
                print()
                print(" **** " +userName + " won ****")
            win = 1
        break
    elif theBoard['4'] == theBoard['5'] == theBoard['6'] != ' ': # across the middle
        printBoard(theBoard)
        if turn == 'O':
            print("\nGame Over. Mr.Elfo won.\n")
            print("Unfortunately you won't get any additional candles.")
            win = 0
        else:
            print()
            print(" **** " +userName + " won ****")
            win = 1
        break
    elif theBoard['7'] == theBoard['8'] == theBoard['9'] != ' ': # across the bottom
        printBoard(theBoard)
        if turn == 'O':
            print("\nGame Over. Mr.Elfo won.\n")
            print("Unfortunately you won't get any additional candles.")
            win = 0
        else:
            print()
            print(" **** " +userName + " won ****")
            win = 1
        break
    elif theBoard['1'] == theBoard['4'] == theBoard['7'] != ' ': # down the left side
        printBoard(theBoard)
        if turn == 'O':
            print("\nGame Over. Mr.Elfo won.\n")
            print("Unfortunately you won't get any additional candles.")
            win = 0
        else:
            print()
            print(" **** " +userName + " won ****")
            win = 1
        break
    elif theBoard['2'] == theBoard['5'] == theBoard['8'] != ' ': # down the middle
        printBoard(theBoard)
        if turn == 'O':
            print("\nGame Over. Mr.Elfo won.\n")
            print("Unfortunately you won't get any additional candles.")
            win = 0
        else:
            print()
            print(" **** " +userName + " won ****")
            win = 1
        break
```

```
elif theBoard['3'] == theBoard['6'] == theBoard['9'] != ' ': # down the right side
    printBoard(theBoard)
    if turn == 'O':
        print("\nGame Over. Mr.Elfo won.\n")
        print("Unfortunately you won't get any additional candles.")
        win = 0
    else:
        print()
        print(" **** " +userName + " won ****")
        win = 1
    break
elif theBoard['3'] == theBoard['5'] == theBoard['7'] != ' ': # diagonal
    printBoard(theBoard)
    if turn == 'O':
        print("\nGame Over. Mr.Elfo won.\n")
        print("Unfortunately you won't get any additional candles.")
        win = 0
    else:
        print()
        print(" **** " +userName + " won ****")
        win = 1
    break
    elif theBoard['1'] == theBoard['5'] == theBoard['9'] != ' ': # diagonal
        printBoard(theBoard)
        if turn == 'O':
            print("\nGame Over. Mr.Elfo won.\n")
            print("Unfortunately you won't get any additional candles.")
            win = 0
        else:
            print()
            print(" **** " +userName + " won ****")
            win = 1
        break

# If neither X nor O wins and the board is full, we'll declare the result as 'tie'.
if count == 9:
    print("It's a Tie!!")
    print("Unfortunately you won't get any additional candles.")
    win = 0
# Change the player after every move.
if turn =='X':
    turn = 'O'
else:
    turn = 'X'
```

```
if win == 1:
    lives += 2
    print("%s has received 2 more candles from Mr.Elfo." %userName)
print("This means you currently have",lives,"candles." )


Click()
print("Well that was a good game to pass some time. Seems like the teleportation is ready to go! ")
Click()
print("Oh here is a little you might want to know about Mister E.")
Click()
#####################################################################################



#land 2 intro
print("\nMister E. was an old friend of your father's, maybe he has some clues about the legacy your father left behind.")
print("Unfortunately, he doesn't like accepting many visitors on his mountain, so he has set a couple of riddles along your path to the summit.")
print("\nWish you the best of luck.")
print()
Click()
```

```
print("\n")
print("Teleported to Mister E's Riddle Mountain...")
print("(Tip: In this stage, you may use the phrase: 'hint' to get help.)")
print("\n")
Click()

#Mister E's Riddle Realm
print("\nYou've arrived at the base of the mountain, but now you are lost.")
Click()
print()
print("As you are looking around, Mister E's messenger pigeon delivers a slip of paper.")
print("It reads...\n")
Click()
print("Hello %s,\nI knew you would someday pay me a visit.\nYour father and I have
travelled together for many years solving puzzles and uncovering mysteries.\nI'm sure
you have many questions for me, but first...\n" %userName)
print("Let's see if you have what it takes to be an adventurer just like your father.\n")
print("Best regards,\nMister E.\n")
print("P.S. Along with this letter, I have enclosed one more item that will help you find
me.")
Click()

#r1
```

```
for i in range(lives + 1):
    q1 = input("Mister E's first guide has lakes with no water, mountains with no stone,
and cities with no buildings. What is it?\n\nAnswer: ")
    if q1.lower() == "map":
        print("CORRECT. Now you are set to go.")
        l2complete += 1
        break
    elif q1.lower() == "hint":
        print("Hint: _ _ _")
    else:
        lives -= 1
        print("INCORRECT. I will break one of your candles. You have now %d candle(s)."
% lives)
    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()

Click()
print("\nAs you start your journey, you realized that this would be a really long and lonely
trip, so you called an Uber driver nearby.")
print()
```

```
#r2
for i in range(lives + 1):
    q2 = input("Once he arrives, you guys continue your trip going the opposite way up a one way road, but then you pass two policemen along the way, none of which stopped you. Why not?\n\nAnswer: ")
    if q2.lower() == "walking" or q2.lower() == "walk":
        print("CORRECT. The Uber driver decided to accompany you until you reach Mister E's place.")
        l2complete += 1
        break
    elif q2.lower() == "hint":
        print("Hint: You guys are ' _ _ _ _ ing'.")
    else:
        lives -= 1
        print("INCORRECT. I will break one of your candles. You have now %d candle(s).\n" % lives)
    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye." %userName)
        sys.exit()
```

```
Click()
print("Uber driver buddy: Dude, I'm so sorry. My car broke down a while ago.(╥̆﹏̆╥)")
print("%s (you): (－‸ლ)" %userName)
print("Uber driver buddy: Well, let's just travel together. (T-T)ヾ(^^ )")
Click()
print("Not too long after, a driver pulls up and ask if you and the Uber driver wanted to
hitch a ride.")
print()

#r3
for i in range(lives + 1):
   q3 = input("In the standard 5-seated car, there are two fathers and two sons in the
car, yet there are enough seats for both you and the Uber driver. How? \n \nAnswer: ")
   if q3.lower() == "grandfather, father, and son" or q3.lower() ==  "grandfather, father,
son" or q3.lower() == "grandfather,father,son" or q3.lower() == "grandfather, father, and
son" or q3.lower() == "grandfather,father,andson" or q3.lower() == "grandfather,father,
andson":
      print("CORRECT. Gladly, you both hop on the car. (o^ω^)(^ω^o)")
      l2complete += 1
      break
   elif q3.lower() == "hint":
      print("Hint: They are '_ _ _ _ _ _ _ _ _ _ _,' ' _ _ _ _ _ _,' and ' _ _ _'.")
   else:
        lives -= 1
        print("INCORRECT. I will break one of your candles. You have now %d
candle(s).\n" % lives)
    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()


Click()
print("Just as the sun was about to set, the car stops...")
Click()
print(" ╰(*'▽')ノ━☆ﾟ.*･｡ﾟ You've finally reached the top!  ")
print()



#r4
for i in range(lives + 1):
    q4 = input("At the summit, you see something that possesses a halo of water, walls of
stone, and a tongue of wood. What do you see?\n\nAnswer: ")
    if q4.lower() == "a castle" or q4.lower() == "castle":
        print("CORRECT. Welcome to Mister E's humble abode.")
        l2complete += 1
        break
    elif q4.lower() == "hint":
        print("Hint: a '_ _ _ _ _ _' ")
    else:
        lives -= 1
        print("INCORRECT. I will break one of your candles. You have now %d
candle(s).\n" % lives)
    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()


Click()
print("Step by step, you cross the drawbridge, searching for a glimpse of Mister E.")
Click()
print("Just as you approached the center of the open courtyard,\nMister E's butler
appears and regretfully informs you of Mister E's passing.")
print("His last words were to pass one last riddle to the child of his good friend, Dr.
Robinson.")


#r5
for i in range(lives + 1):
    q5 = input("What are the next three letters in this combination? OTTFFSS_ _ _\n\nAnswer: ")
    if q5.lower() == "ent" or q5.lower() == "e n t":
        print("CORRECT. Now you have unlocked the passcode to a secret chamber that will directly transport you to the heart of ENTertainment.")
        l2complete += 1
        break
    elif q5.lower() == "hint":
        print("Hint: think of counting numbers and its relationship to the combo.")
    else:
        lives -= 1
        print("INCORRECT. I will break one of your candles. You have now %d candle(s).\n" % lives)
    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye." %userName)
        sys.exit()

#transition to land 3
if l2complete == 5:
    gameMap["Mister E's Riddle Realm"] = "complete"
```

```
print("\n\n")
click = input("Hello %s. Congratulations! You have conquered Mister E's Riddle Realm.
Look at your map, it has updated.\n\n\n(press Enter to continue)" %userName)
print()
checkClick(click)
ProgressPrint(gameMap)


Click()
print()
print("The next stop on your journey is at %s." % gameIsland[2])
if lives < 3:
   lives = 3
   print("\nYour candles have been restocked back to %d." % lives)
else:
   print("You still have %d candles remaining." %lives)
print()
Click()



#land 3 intro
print()
print("Welcome to Fame of Us Island! You have qualified to visit this game show like island.\nNo this isn't Survivor, you will have to answer the following pop culture related questions.")
print("\nI will be your host: Steve Harvey ( ͡ᵔ ͜ʖ ͡ᵔ )")
print("Wish you the best of luck. -Game Show Music Begins Playing-")
print()
Click()
print()
print("Teleported to %s..." % gameIsland[2])
print()
Click()

#Fame of Us Island
#q1
q1 = MC("Question 1: Who voiced Shrek the ogre in the internationally-acclaimed award-winning motion picture shrek?", "A")
q1.addChoice("A. Mike Myers")
q1.addChoice("B. Eddie Murphy")
q1.addChoice("C. Toby Mcguire")
q1.addChoice("D. Peter Griffin")

while lives > 0:
    print("\n")
```

```
  q1.display()
  userAns = input("Your answer: ")
  q1.checkAnswer(userAns.upper())
  if userAns.upper() == q1.answer:
    l3complete += 1
    break
  else:
    lives -= 1
    print("I will break one of your candles. You have now %d candle(s)" %lives)
  if lives == 0:
    print("\nGAME OVER. Steve Harvey's facial expression: ಠ_ಠ")
    print("You are out of candles. %s, you have failed your mission. Goodbye." %userName)
    sys.exit()

#q2
q2 = MC("Question 2: Complete the following phrase: Scooby Dooby Doo __________", "A")
q2.addChoice("A. Where are you")
q2.addChoice("B. Who are you")
q2.addChoice("C. I love you")
q2.addChoice("D. I see you")
```

```
while lives > 0:
    print("\n")
    q2.display()
    userAns = input("Your answer: ")
    q2.checkAnswer(userAns.upper())
    if userAns.upper() == q2.answer:
        l3complete += 1
        break
    else:
        lives -= 1
        print("I will break one of your candles. You have now %d candle(s)" %lives)
    if lives == 0:
        print("\nGAME OVER. Steve Harvey's facial expression: ಠ_ಠ")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()

#q3
q3 = MC("Question 3: What movie director attended and dropped out of California State
University Long Beach?", "B")
q3.addChoice("A. Micheal Bay")
q3.addChoice("B. Steven Spieldberg")
q3.addChoice("C. Quentin Terantino")
```

```
q3.addChoice("D. Will Smith")

while lives > 0:
    print("\n")
    q3.display()
    userAns = input("Your answer: ")
    q3.checkAnswer(userAns.upper())
    if userAns.upper() == q3.answer:
        l3complete += 1
        break
    else:
        lives -= 1
        print("I will break one of your candles. You have now %d candle(s)" %lives)
    if lives == 0:
        print("\nGAME OVER. Steve Harvey's facial expression: ಠ_ಠ")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()
#q4
q4 = MC("Question 4: Who is internationally known for their stylish basketball shoes?",
"A")
q4.addChoice("A. Micheal Jordan")
q4.addChoice("B. Micheal Jackson")
```

```
q4.addChoice("C. Mick Jagger")
q4.addChoice("D. O.J. Simpson")

while lives > 0:
  print("\n")
  q4.display()
  userAns = input("Your answer: ")
  q4.checkAnswer(userAns.upper())
  if userAns.upper() == q4.answer:
    l3complete += 1
    break
  else:
    lives -= 1
    print("I will break one of your candles. You have now %d candle(s)" %lives)
  if lives == 0:
    print("\nGAME OVER. Steve Harvey's facial expression: ಠ_ಠ")
    print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
    sys.exit()
#q5
q5 = MC("Question 5: Who is known as one of the greatest and influential
skateboarders in the world?", "D")
q5.addChoice("A. David Bowie")
```

```
q5.addChoice("B. Lebron James")
q5.addChoice("C. Heath Ledger")
q5.addChoice("D. Tony Hawk")


while lives > 0:
    print("\n")
    q5.display()
    userAns = input("Your answer: ")
    if userAns.upper() == q5.answer:
        print("CORRECT!")
        l3complete += 1
        break
    else:
        lives -= 1
        print("I will break one of your candles. You have now %d candle(s)" %lives)
    if lives == 0:
        print("\nGAME OVER. Steve Harvey's facial expression: ಠ_ಠ")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()

######################################################################
########
```

```
#NEW ADDITION: WhoMan
Click()
print("Game show producer: Psst... Over here.")
print("I've got another exclusive prize here, but it not a listed prize in the show.")
Click()
print("I'll make you a deal. If you can guess the name of this incredible legendary person, I'll give you the prize.")
print("But here's the catch..., you can only guess 7 times.")
Click()
print("Let's begin the game of WhoMan, basically Hangman (just that you are guessing a specific human)")
Click()


turns = 7
def showHangman(turns):
    if turns == 7:
        print(" --------")
        print(" |      |")
        print(" |      ")
        print(" |     ")
        print(" |      ")
        print(" |     ")
        print(" -")
```

```
elif turns == 6:
    print(" --------")
    print(" |      |")
    print(" |      O")
    print(" |      ")
    print(" |      ")
    print(" |     ")
    print(" -")
elif turns == 5:
    print(" --------")
    print(" |      |")
    print(" |      O")
    print(" |      |")
    print(" |      ")
    print(" |     ")
    print(" -")
elif turns == 4:
    print(" --------")
    print(" |      |")
    print(" |      O")
    print(" |     \\|")
    print(" |      ")
    print(" |     ")
    print(" -")
elif turns == 3:
    print(" --------")
    print(" |      |")
    print(" |      O")
    print(" |     \\|/")
    print(" |      ")
    print(" |     ")
    print(" -")
elif turns == 2:
    print(" --------")
    print(" |      |")
    print(" |      O")
    print(" |     \\|/")
    print(" |      |")
    print(" |     ")
    print(" -")
elif turns == 1:
    print(" --------")
    print(" |      |")
    print(" |      O")
    print(" |     \\|/")
    print(" |      |")
```

```
        print(" |     / ")
        print(" -")
    elif turns == 0:
        print(" --------")
        print(" |      |")
        print(" |      O")
        print(" |     \\|/")
        print(" |      |")
        print(" |     / \\")
        print(" -")



word = "STANLEE"
guesses = ''
guessSet = set(guesses)
currentfailed = len(word)
currentSetlen = len(guessSet)
whoMan = 0
print("Clues: The first name is 4 letters. The last name is 3 letters.")
print("Start guessing...")
while turns > 0:
    item = ''
```

```
outputline = ''
failed = 0 #local variable reset each round
guess = input("guess a character: ")
print()
guess = guess.upper() #your input each round
guesses += guess
for item in word: #check letter in each index of word
    if item in guesses:
        print(item)
        outputline += (item)
        outputline += (' ')
    else:
        print("_")
        outputline += ("_")
        outputline += (" ")
        failed += 1
print()
print(outputline)
if failed == 0:
    print("\nYou won, Superman.") #Won
    whoMan = 1
    break
    guessSet = set(guesses)
    print("Current guesses:")
    print(guessSet)
    if failed == currentfailed and len(guessSet) > currentSetlen: #if guess is wrong from a
new letter
        turns -= 1
        print("Wrong")
        Click()
        showHangman(turns)
    elif failed == currentfailed and len(guessSet) == currentSetlen: #if guess is repeated
        print("You've already tried this.")
    else:
        print("Correct")
    print("You have", + turns, 'more attempts')
    currentfailed = failed
    currentSetlen = len(guessSet)
#   print("failed:", failed, "currentfailed:", currentfailed)
    if turns == 0:
        print("You Lose")#Lost
Click()
if whoMan == 1 and word == 'STANLEE':
    print("Here's your prize....'drum roll'")
    Click()
    print("Stan Lee's autograph on the first issue of 'Amazing Fantasy No. 15' where Spiderman was first debuted.")
    print("Actually, this is your father's special birthday gift to you on the condition that you passed this test.")
if whoMan == 0 and word == 'STANLEE':
    print("It's a bummer you lost, but don't fret.\nYou still passed this island.")
    print("Just no bonus prizes. ╮(╯ _╰ )╭")
Click()

##################################################################



#transition to land 4


if l3complete == 5:
    gameMap["Fame of Us Island"] = "complete"


print("\n\n")
click = input("%s. Woah congratulations! You're the smartest contestant we have had since Albert Einstein was on the show back in 1945.\n\nSteve Harvey's facial expression: (╯°□°)╯ \n\n-Game Show Outro Begins To Play-\nYou have conquered Fame of Us Island. Look at your map, it has updated.\n\n\n(press Enter to continue)" %userName)
```

```
print("\n")
checkClick(click)
ProgressPrint(gameMap)

print()
Click()
print()
print("The next stop on your journey is at %s." % gameIsland[3])
if lives < 3:
   lives = 3
   print("\nYour candles have been restocked back to %d." % lives)
else:
   print("You still have %d candles remaining." %lives)
print()
Click()

#land 4 intro
print("\n\n")
#print("Sum Divided Forest intro...")
print("Wish you the best of luck.")
print("\n")
Click()
print()
print("Teleported to %s..." % gameIsland[3])
print()
Click()

#Sum Divided Forest
print("Well well well, look who we have here. Its Dr. Robinson's son... What's the name? What's the name again?")
print("It's %s right?" %userName)
Click()
print()
print("\nO my my, I've done it again... I'VE DONE IT AGAIN! I am once again.\nThee genius of the world! AHAHHAHA. No one can outsmart me and my questions!")
print()
print("Now %s, the success rate to my questions are a slim to none. With your father only being the one to beat them." %userName)
Click()
print("Which means, I am smarter than 99.99% of the WORLD!!")
print("But for you, I will give you an opportunity to gain additional candles for the first two questions... since I like your father.")
Click()
print("Are you scared??... Are you ready??... Of course you're not. You probably can't even get through my first question!")
Click()
```

```
#q1
#family problem

fam1 = [85, 60, 84, 77, 71]
sumFam = int(sum(fam1))
avgFamInt = int(sumFam)/len(fam1)
avgFamRound = round(avgFamInt)
avgFam = int(avgFamRound)

print("Question 1: *ahem* probably your only question AHAHAHAHAHA!")          #lol#
print("These are the height of my family in inches. Not FEET, but in inches: ")
print(fam1)

uSumFam = ""
while uSumFam != sumFam or uSumFam == ValueError and lives > 0:
    try:
        uSumFam = input("\nWhat is our total combined height in inches? HMMMM?
\nAnswer:")
        uSumFam = int(uSumFam)
        if uSumFam == sumFam:
            lives += 1
            print("\nOh Oh my. That umm, that is correct. I shall give you one additional
candle. You have a total of %d candles." %lives)
            l4complete += 1
            break
        elif uSumFam != sumFam:
            print("\nAhh yes! No one can defeat the genius thee Mr. Pi's questions!")
            lives -= 1
            if lives == 0:
                print("Well, what did I tell you %s." %userName, "It was obvious that you
couldn't pass my questions... just like the other 99.99%. AHHAHAHA. Goodbye now.")
                print("\nGAME OVER")
                sys.exit()
        if lives > 0:
            print("You do have %d candles remaining though. So good luck... You're going to
need them...\n" %lives)
            #Click()
    except ValueError as exception:
        print("Error:", str(exception), "\n")


if lives == 0:
    print("Well, what did I tell you %s." %userName, "It was obvious that you couldn't
pass my questions... just like the other 99.99%. AHHAHAHA. Goodbye now.")
    print("\nGAME OVER")
    sys.exit()
Click()



#q2: Family problem p2
uAvgFam = ""
print("Well, you managed to get the first question right. I like you... Here is another
opportunity to gain another candle before my MAIN MASTERPIECE!")
while uAvgFam != avgFam or uAvgFam == ValueError and lives > 0:
    try:
        uAvgFam = input("\nNow... What is the average height of my family in inches?
(Round to the nearest whole number) \nAnswer: ")
        uAvgFam = int(uAvgFam)
        if uAvgFam == avgFam:
            lives += 1
            print("That is correct! You have gained 1 additional candle! You have a total of
%d candles." %lives)
            l4complete += 1
            break
        elif uAvgFam !=avgFam:
            print("That is not the right answer.")
            lives -= 1
            if lives == 0:
                print("Well, what did I tell you %s." %userName, "It was obvious that you
couldn't pass my questions... just like the other 99.99%. AHHAHAHA. Goodbye now.")
                print("\nGAME OVER")
                sys.exit()
        if lives > 0:
            print("You have %d candle(s) remaining. Please use them wisely and try
again.\n" %lives)
            #Click()
    except ValueError as exception:
        print("Error:", str(exception), "\n")

Click()


# if lives == 0:
#     print("Well, what did I tell you %s. " %userName, "It was obvious that you couldn't
pass my questions... just like the other 99.99%. AHHAHAHA. Goodbye now.")
#     print("\nGAME OVER")
#     sys.exit()


if lives > 3:
    print("\nHmm... I guess you're smarter than I thought. Now you're going into my
Masterpiece with more lives.\n")
    Click()
```

```
elif lives <=3:
    print("\nAlthough you didn't manage to go into the final question with more than 3 candles, I hope %d candles should be enough.\n" % lives)
    Click()

print("\nHere is the last and final problem. I will be generating a number between 1-10. You need to figure out what that number is.\n\nDo not worry, if you lose a candle, I will give you a hint.\nI only have 3 hints so use them wisely.\n")
Click()

#q3
import random
r = random.randint(1,10)
count = 0

print("\nPlease consider that ANY answer you put will count so please follow the rules. Or Else.\n")
Click()

print()
print("Pick a number between 1-10.")
if r%2 == 0:
    print("Here is your first hint: The number is even.")
```

```
else:
    print("Here is your first hint: The number is odd.")
userInput = ""
while userInput != r or userInput == ValueError and lives > 0:
    try:
        userInput = input("What is your answer? \nAnswer:")
        userInput = int(userInput)
        if userInput == r:
            print("You are correct! The correct number was %d." %r)
            l4complete += 1
            break
        elif userInput != r:
            print("That is not the right answer.")
            lives -= 1
            count += 1
            print("You have %d candle(s) left.\n" %lives)
            #Click()
    except ValueError as exception:
        print("Error:", str(exception), "\n")

    if count == 1:
        print("Here is your second hint:")
        if r > 0 and r < 5:
```

```
        print("The number is in the lower half.")
    elif r == 5:
        print("the number is neither in the lower or upper half.")
    elif r > 6:
        print("The number is in the upper half.")
if count == 2:
    if r == 1:
        print("This number shows the best of the best.")
    if r == 2:
        print("This number is very photogenic.")
    if r == 3:
        print("This number likes to crash on dates.")
    if r == 4:
        print("This number likes to receive dates.")
    if r == 5:
        print("This number is a handful.")
    if r == 6:
        print("This number is very lonely on two hands.")
    if r == 7:
        print("This number is the most popular number.")
    if r == 8:
        print("This number, if on its side, is the biggest of all numbers.")
    if r == 9:
            print("This number is fine and likes wine.")
        if r == 10:
            print("This number is the first to be said on new years eve.")

    if lives == 0:
        print("\nGAME OVER")
        print("You are out of candles. %s, you have failed your mission. Goodbye."
%userName)
        sys.exit()

#transition to land 5 (CLOSING STAGE)
if l4complete == 3:
    gameMap["Sum Divided Forest"] = "complete"
    print("Congratulations! You have completed the last land.")

print()
click = input("Hello %s. Congratulations! You have conquered Sum Divided Forest. Look
at your map, it has updated.\n\n\n(press Enter to continue)" %userName)
print()
checkClick(click)
ProgressPrint(gameMap)
Click()
print()
print("You completed the quest your father left behind.")
print()
Click()
print("Now I can present to you this box that your father has left behind.")
click = input("Click Enter to open the box...")
print("*open*\n")
DadLetterW()
DadLetterR()


Click()
print("Well %s... If you do decide to follow your dad, the door to the next universe is right behind me." %userName)
Click()
print("*Opening Door*")
Click()
print("Farewell %s, and good luck with your journey." %userName)
print(" *Enter through the door*")
print("Fin... For now")


print("\n")
print("?arallel Quest: Created by Catherine Dao, Hang Le, Kevin Pham, Robert Torres")
```