Project Report

Creating a Hangman game using python

Submitted by
Atthaluri Shashank
M015906

Under the guidance of
Dr. Ajit Kumar
Institute of Chemical Technology, Mumbai

Centre for Excellence in Basic Sciences
University of Mumbai
Kalina, Mumbai, India – 400098

CONTENTS

Topics

- Introduction
- What is meant by Hangman?
- Program flowchart
- Description of program step-wise
- Source code
- How the program runs
- Output
- Reference

Title of the project: Hang man (A Multi-player game)

Pattern of the game: Playing one after the other by guesssing the word chosen randomly

Minimum requirements: Python 2.7 version in the system

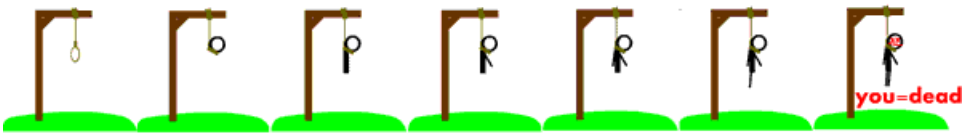
Introduction

Python is a widely used high-level programming language for general-purpose programming.Basically we are using this language for coding to create a game. We use normally pygame to create games using through the python language. Here are the list of games made in python

- snake game
- tetris
- Tic Tac Toe
- Big games like Battlefield and Disney's Toontown is written in Python
- Civilization IV uses Python for most of its tasks.
- Freedom Force,Frets on Fire,Vampire: The Masquerade Bloodlines.

Coming to our program, it is a multi-player game of hang man.In this program we imported things which were useful at different situations. As we import random. Basically the game starts with the below steps:-

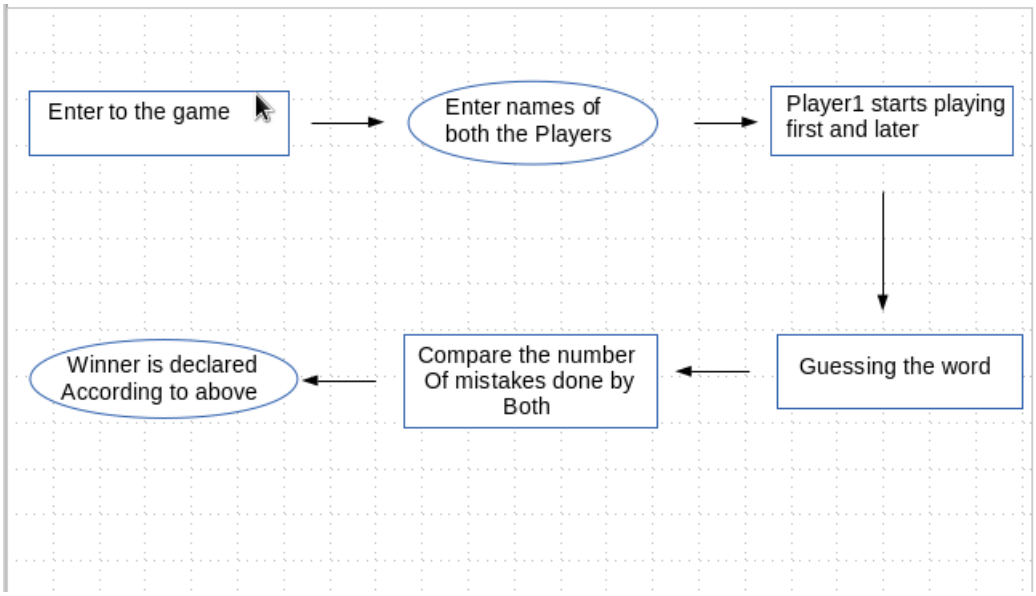
- First the player1 starts playing later player2.
- guessing the word given by system randomly.
- After guessing the word by player1 the score is stored and player2 starts his game.
- After player2 completes the game both the scores were compared and declare the result.



What is meant by Hangman?

Hangman is a paper and pencil guessing game for two or more players. One player thinks of a word and the other tries to guess it by suggesting letters, within a certain number of guesses.
The word to guess is represented by a row of dashes, representing each letter of the word. In most variants, proper nouns, such as names, places, and brands, are not allowed. If the guessing player suggests a letter which occurs in the word, the other player writes it in all its correct positions. If the suggested letter does not occur in the word, the other player draws one element of a hanged man stick figure as a tally mark.

The player guessing the word may, at any time, attempt to guess the whole word. If the word is correct, the game is over and the guesser wins. Otherwise, the other player may choose to penalize the guesser by adding an element to the diagram. On the other hand, if the other player makes enough incorrect guesses to allow his opponent to complete the diagram, the game is also over, this time with the guesser losing. However, the guesser can also win by guessing all the letters that appears in the word, thereby completing the word, before the diagram is completed.
As the name of the game suggests, the diagram is designed to look like a hanging man..The exact nature of the diagram is players draw the gallows before they begin the game and draw parts of the man's body (traditionally the head, then the torso, then the arms and legs one by one).



Note:Above is the game flowchart

Description of the program:

Step1:

This step contains the file that is used in the whole program. In this program we have imported different codes like random. Random is to give the random word as a challenge to be guessed by the palyers.

Step2:

The game starts with input of players names and also the level of the game. Next, Rules of the game is displayed, player1 starts the game followed by player2.

Step3:

Player1 starts guessing the word letter by letter given by the computer.If the player guesses the letter correctly the point increase by 1 and if he/she enters the letter not present in the word, one point is deducted.

Step4:

If player1 either wrongly guesses 6 times or correctly guesses the word within the 6 times limit then he/she lose/wins the round respectively.The game is taken over by player2. This process continues.

Step5:

If there is tie between player1 and player2, then the number of steps taken by each player is considered for declaring the winner.

Source code

- Code of the program

```
import random
#welcoming the users
name1 = raw_input("What is your name? ")
name2 = raw_input("What is your name? ")
print "Hello, " + name1, "Time to play hangman!"
print "Hello, " + name2, "Time to play hangman!"
print "You have a chance to choose the level, Press (1) for easy level.\n(2)
#input of the level
user_choice_1 = raw_input("->")

player1_score=0
player2_score=0
#Rules of the game
print "Rules of the game"
print "Basically you should save yourself out from this trap."
print "What you can do is just follow the rules below."
print "1)The system will give u some blanks which is a word not revealed."
print "2)you should guess that word correctly."
print "3)the number of moves lead you to be get hanged."
print "are you ready to die?"
print "First,"+ name1, "starts playing,"+ name2, "try to distrub him to win"
#Here starts the game
print "Start guessing..."
#list of word for easy level
wordlist1 = ['hangman','dinner','computer','america','olympics','oscar',
          'football','minecraft','jacket','cabbage','electricity',
            'pasta','japan','water','programming','anaconda','dog',
            'name','windows','curtains','wheel','civilization',
            'bluebird','table','size','guardian','mario','parachute',
            'bioshock','physics','jumping','eating','uranium','obama',
            'youtube','putin','dairy','christianity','coins','heard',
            'mom','executive','car','jade','abrahim','sand','silver',
            'fizzle','moonman','watermelon','whistle','hesitate','agario',
             'extreme','wrist','damn','shit','wrong','turtle','church',name1]
```

```
#list of words for hard level
wordlist2 = ['wry', 'hymn','cyst', 'myrrh', 'myth', 'wyrm', 'crypt',
            'flyby', 'gypsy','lynch', 'nymph', 'pygmy', 'rhythm',
            'hurdler','purpler','weird','playright','agitating',
            'kirito','montenegro','clubpenguin','chlorinate',
              'oskahlavistah','dermatoglyphics','misconjugatedly',
              'uncopyrightable','duplicate','abridgment','clustering',
              'angleworms','authorized','algorithms','benchmarks',
              'bifurcated','blacksmith','boundaries','championed',
              'complexity','consumable','crumbliest','deathblows',
              'demography','emulations','exhausting','falterings',
              'forgivable','fulminated','graciously','harlequins',
              'incubator','journalism','labyrinths','lawrencium', name1]

#game starts according to user choice
if   user_choice_1=='1' :

        #here we set the secret
        words = random.randint(0, len(wordlist1))
        word1 = (wordlist1[words])


elif user_choice_1=='2' :

        #here we set the secret
        words = random.randint(0, len(wordlist2))
        word1 = (wordlist2[words])

#creates an variable with an empty value
guesses1 = ''

#determine the number of turns
turns1 = 6

# we Create a while loop

#check if the turns are more than zero
while turns1 > 0:
```

```
# make a counter that starts with zero
failed1 = 0

# for every character in secret_word
for char1 in word1:

# see if the character is in the players guess
    if char1 in guesses1:

    # print then out the character
        print char1,

    else:

    # if not found, print a blank
        print "_",

    # and increase the failed counter with one
        failed1 += 1

# if failed is equal to zero

# print You guessed correctly
if failed1 == 0:
    print "Congrats You guessed correctly, Lets see who is going to be th

# exit the script
    break

print

# ask the user go guess a character
guess1 = raw_input("guess a character:")

# set the players guess to guesses
guesses1 += guess1
```

```
# set wrong if player enter anything other than alphabets
if guess1.isalpha() == False:
    print "You have to guess a letter, silly!. BTW you lost a chance sorr

# set wrong if the input is more then one letter
if len(guess1) > 1:
    print "You can't guess more than one letter at a time, silly!. Btw yo




# if the guess is not found in the secret word
if guess1 not in word1:

 # turns counter decreases with 1 (now 5)
    turns1 -= 1

# prints wrong guess
    print "Things aren't looking so good, that guess was WRONG!"

# how many turns are left
    print "You have", + turns1, 'more guesses1'


    if turns1 == 5:
         print "-------"
         print "|      |"
         print "|      0"
         print "|"
         print "|"
         print "|"
         print "----------"
```

```
if turns1 == 4:
     print "-------"
     print "|      |"
     print "|      0"
     print "|      |"
     print "|"
     print "|"
     print "----------"

if turns1 == 3:
     print "-------"
     print "|      | "
     print "|      0 "
     print "|     <|"
     print "|"
     print "|"
     print "----------"

if turns1 == 2:
     print "-------"
     print "|      |"
     print "|      0"
     print "|     <|>"
     print "|"
     print "|"
     print "----------"

if turns1 == 1:
     print "-------"
     print "|      |"
     print "|      0"
     print "|     <|>"
     print "|     / "
     print "|"
     print "----------"
```

```
        if turns1 == 0:
             print "-------"
             print "|      |"
             print "|      0"
             print "|     <|>"
             print "|     / /"
             print "|"
             print "----------"

         # if the turns are equal to zero
        if turns1 == 0:


            print "Sorry you didn't guess the word correctly,the word is,"+wo



print "its your turn,"+ name2, "try to defeat,"+name1, "with maximum points"
#Here starts second player  game
print "Start guessing..."
#game starts according to user choice
if   user_choice_1=='1' :

        #here we set the secret
        words = random.randint(0, len(wordlist1))
        word2 = (wordlist1[words])


elif user_choice_1=='2' :

        #here we set the secret
        words = random.randint(0, len(wordlist2))
        word2 = (wordlist2[words])

#creates an variable with an empty value
guesses2 = ''

#determine the number of turns
```

```
turns2 = 6

# Create a while loop

#check if the turns are more than zero
while turns2 > 0:

    # make a counter that starts with zero
    failed2 = 0

    # for every character in secret_word
    for char2 in word2:

    # see if the character is in the players guess
        if char2 in guesses2:

        # print then out the character
            print char2,

        else:

        # if not found, print a blank
            print "_",

        # and increase the failed counter with one
            failed2 += 1

    # if failed is equal to zero


    if failed2 == 0:
        print "Congrats You guessed correctly Lets see who is going to be the

    # exit the script
        break

    print
```

```
# ask the user go guess a character
guess2 = raw_input("guess a character:")

# set the players guess to guesses
guesses2 += guess2

  # set wrong if player enter anything other than alphabets
if guess2.isalpha() == False:
    print "You have to guess a letter, silly!. BTW you lost a chance sorr

# set wrong if the input is more then one letter
if len(guess2) > 1:
    print "You can't guess more than one letter at a time, silly!. BTW yo


# if the guess is not found in the secret word
if guess2 not in word2:

 # turns counter decreases with 1 (now 5)
    turns2 -= 1

# print wrong
    print "Things aren't looking so good, that guess was WRONG!"

# how many turns are left
    print "You have", + turns2, 'more guesses2'

 # picturization of the game
    if turns2 == 5:
         print "-------"
         print "|      |"
         print "|      0"
         print "|"
         print "|"
         print "|"
         print "----------"
```

```
if turns2 == 4:
     print "-------"
     print "|      |"
     print "|      0"
     print "|      |"
     print "|"
     print "|"
     print "----------"

if turns2 == 3:
     print "-------"
     print "|      | "
     print "|      0 "
     print "|     <| "
     print "|"
     print "|"
     print "----------"

if turns2 == 2:
     print "-------"
     print "|      |"
     print "|      0"
     print "|     <|>"
     print "|"
     print "|"
     print "----------"

if turns1 == 1:
     print "-------"
     print "|      |"
     print "|      0"
     print "|     <|>"
     print "|     / "
     print "|"
     print "----------"
```

```
        if turns2 == 0:
             print "-------"
             print "|      |"
             print "|      0"
             print "|     <|>"
             print "|     / /"
             print "|"
             print "----------"


        # if the turns are equal to zero
        if turns2 == 0:


             print "Sorry you didn't guess the word correctly,
             the word is,"+word2
#players score according to their mistakes
player1_score=player1_score + turns2
print player1_score,player2_score
player2_score=player2_score + turns1
if player1_score > player2_score:
        print "Congratulations Mr."+name2, "you won the game"
elif player1_score < player2_score:
        print "Congratulations Mr."+name1, "you won the game"
else :
        print "The game is tie. Play again."
```

Programs runs

- Below figures represents how the program runs in the terminal.
- Caption represents the situation of the terminal.

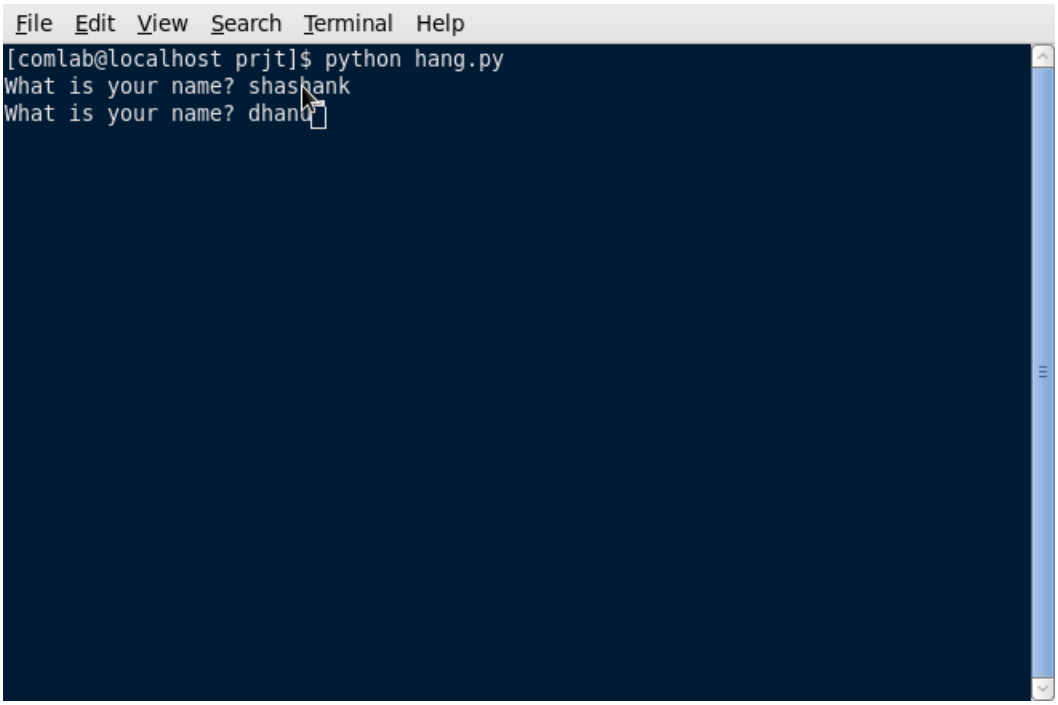


Figure 1:-Players enter their names

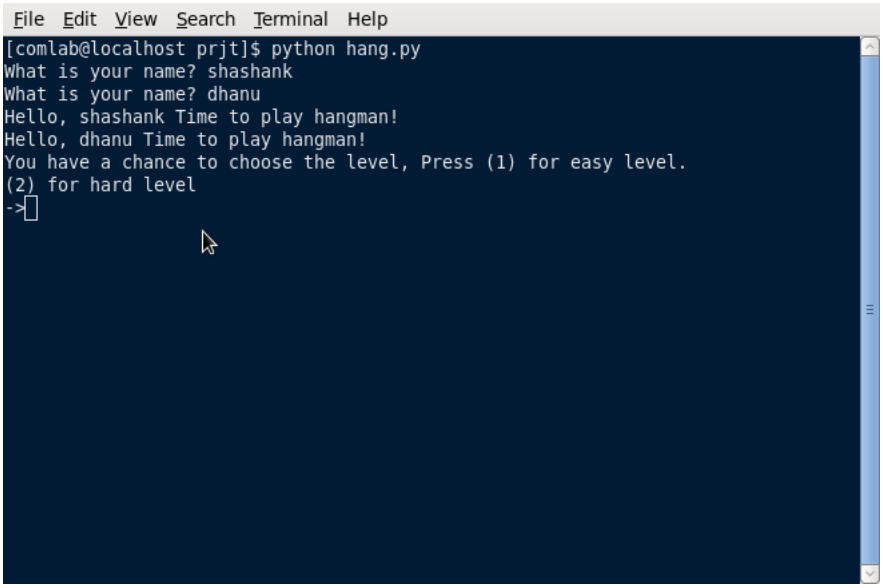


Figure 2:-Choice of level of the game

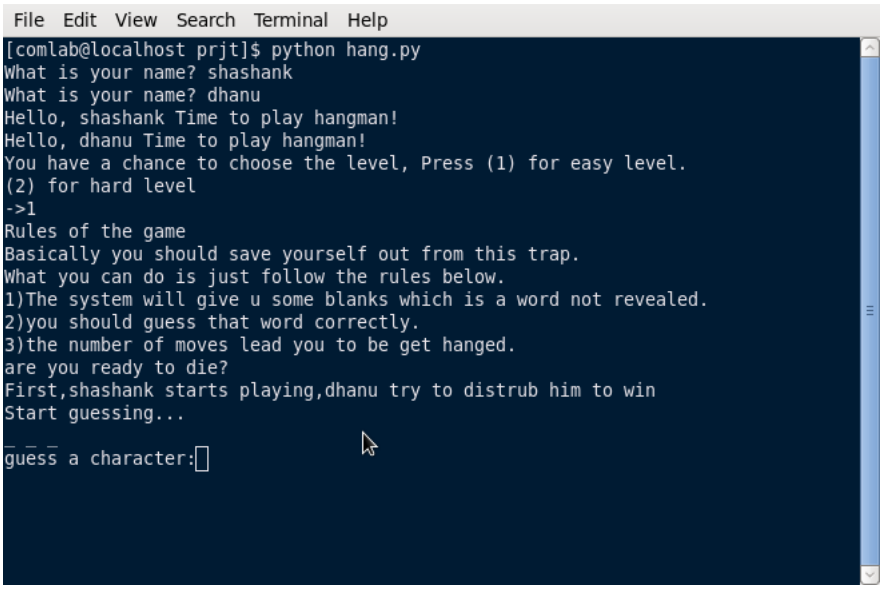


Figure 3:-Rules of the game and player 1 starts playing

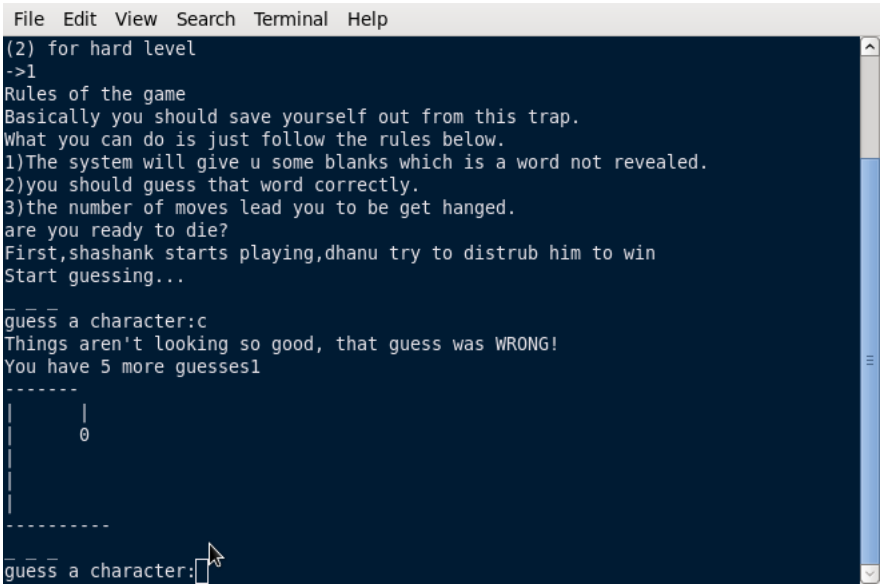


Figure 4:-If the guess is wrong. The diagram represents the number of chances left.

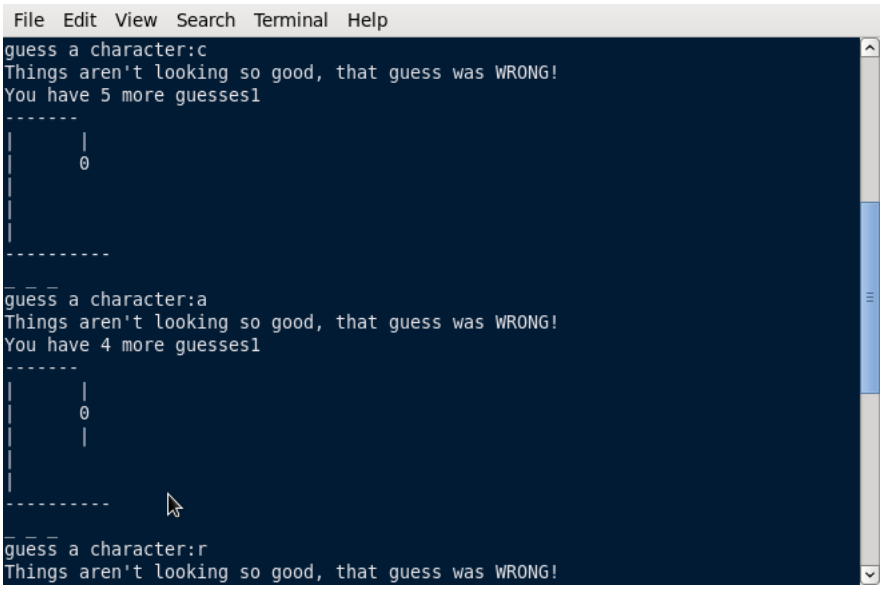


Figure 5:-As the number of guesses get wrong the person is going to hang!

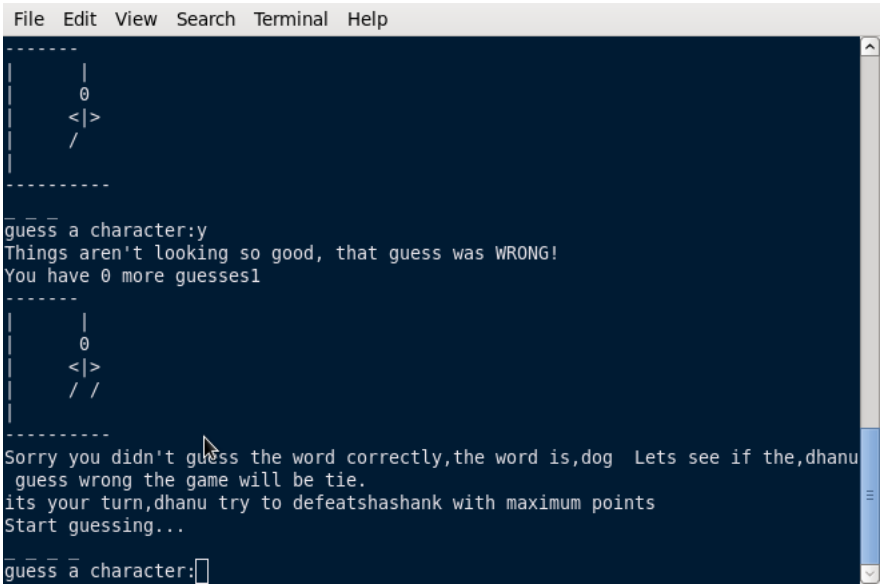


Figure 6:-Player 1 cannot guess correctly and Player 2 starts his game.

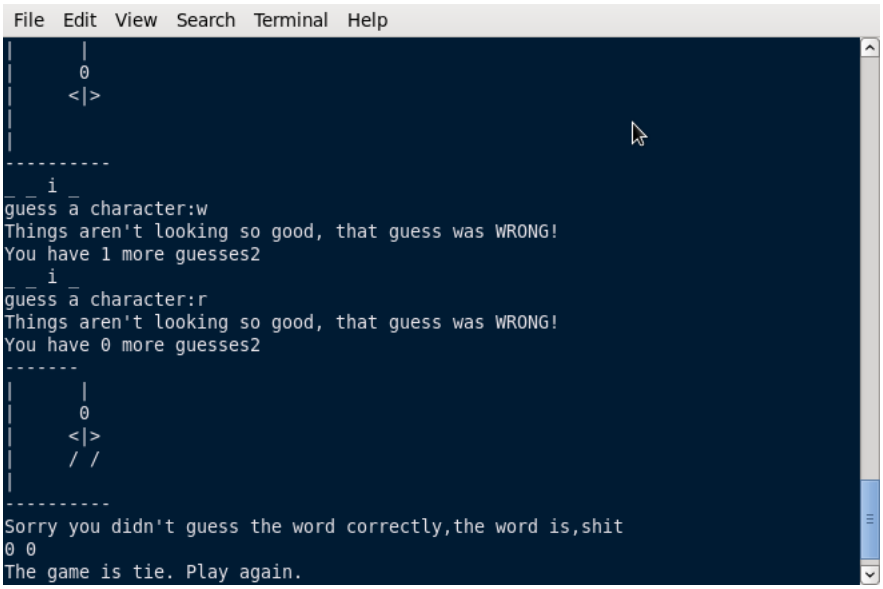


Figure 7:-If the player 2 also cannot guess the word then the game is tie!

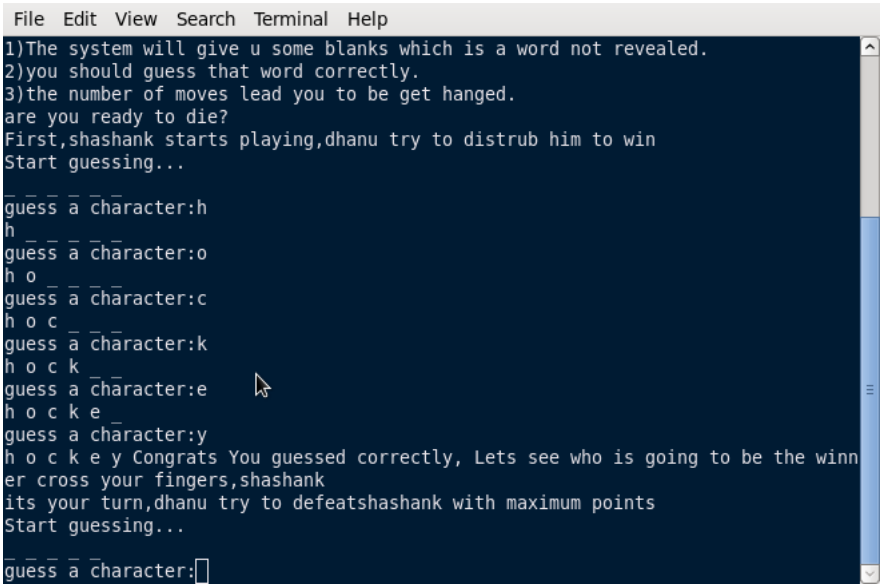


Figure 8:-If the Player 1 guesses his word correctly

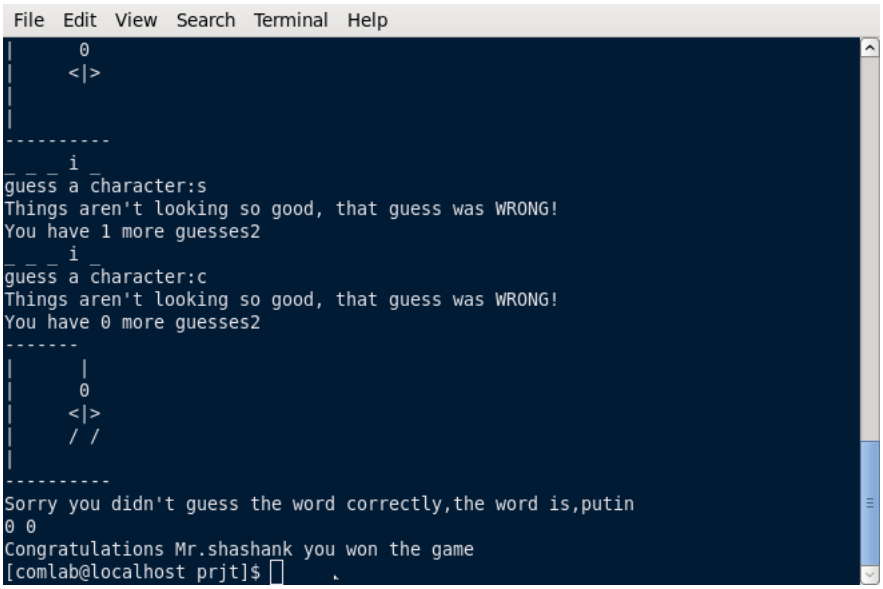


Figure 9:-If Player2 cannot guess the word correctly then the Player 1 won

Output

The winner is decided by the difference over the steps to guess the word and if the number of steps are equal the game is tie.

1 Reference

- Script from wikipedia and some help from my senior.

Things done in report

- Report writing was done in latex.
- Figures were done by using Gwenview.
- Source code is the python program.