CSC-1200 HW4: Hangman

Introduction

You will finish a Python program that plays the game Hangman.

Part 1: Code for Free

You will be given two files. The first is a text file with words to be used in the game. The other is a python file with functions already implemented for you.

Note that the file is not finished.  In fact, there are two functions that are incomplete.  You will finish the functions yourself.

Part 2: Your part

The two functions that you must implement yourself are the blanks\_gone() function and the main() function.  The blanks\_gone() function takes a string as a parameter,  returns either True or False.  It returns True if the string has no underscores in it (the "\_" character).  It returns false if the string has any underscores in it.  Use the string find() function to implement blanks\_gone().  Simply google "python string find" and your will find plenty of descriptions and examples.

Next, you will finish the main() function according to the following algorithm.  Your professor will explain the algorithm in class, as well as the functions given in part 1.

|  |
| --- |
| set **state** to 0 set **losing\_state** to 6  set **sentence** to a random word from the text file. This can be done in a variety of ways. The chosen word will need to be converted to lower using .lower()  set **places** to ""  use a for loop to add a “\_” to **places** for every character in **sentence**  set **winner** to False  loop while **winner**is False and **state** is not the **losing\_state**:         call the draw\_hangman() function passing it the value of **state**          display the **places** string         prompt the user to "Enter a character: "         read the character form the keyboard int the **char** variable         call the replace\_all() function passing it the **sentence**, **places** and **char** variables         set **success**, **sentence**, and **places** to the tuple that replace\_all() returns         if **success** is false           add one to **state**         end if         call blanks\_gone() and pass it the **places** variable         if blanks\_gone() returned True           set **winner** to True         end if end loop  call draw\_hangman() with passing it the value in the **state** variable if **winner** is True       display "CONGRATS! You Win!" else       display "Sorry, you lose :-(" end if |
|  |

**Rubric**

|  |  |
| --- | --- |
| 10 pts | Comments and style (Only your part) |
| 50 pts | Main is correctly implemented using the algorithm above |
| 10 pts | Word is randomly selected from a file |
| 10 pts | Places is correctly set to contain a “\_” for every character in sentence. |
| 20 pts | Blanks\_gone() is correctly implemented |