Comments:

WARNING: If your `project3.py` does not run on the Olin 326 machine on IDLE, the grader will automatically give you a total score of 1 point (for writing the information at the top of your file). You can use a token to resubmit your project one week after the projects are graded.

Otherwise, you will earn one point for each of the following.

1. _____ At the top of your `project3.py` file, you have included your name, credit, and description. *Partial credit* is possible.

   Comment:

2. _____ You have written a correct contract and docstring for the `beats` function. *Partial credit* is possible.

   Comment:

3. _____ You have written a correct contract and docstring for the `isLegal` function. *Partial credit* is possible.

   Comment:

4. _____ Your `isLegal` function does not contain more than one nested selection (if) statement. Your `beats` function does not contain more than one nested selection (if) statement. That is, you may have one if statement contained in another if statement, but no more than that. *Partial credit* is possible.

   Comment:

5. _____ You have written a correct contract and docstring for the `unPigWord` function. You have written a correct contract and docstring for the `pigWord` function. *Partial credit* is possible.

   Comment:

6. _____ Your `unPigWord` function makes good use of Python's string-handling features (Table 3.1 and Table 3.2 of the textbook), avoiding excess complexity. For example, you should not import modules or use technology other than string operators and methods. You do not need a for loop. *Partial credit* is possible.

   Comment:

7. _____ You have written a correct contract and docstring for the `findVowel` function. *Partial credit* is possible.

   Comment:

8. _____ In correctly finding the first vowel within a string, your `findVowel` function makes good use of Python's string-handling features (Table 3.1 and Table 3.2 of the textbook) to avoid excess complexity. You only need one `for` loop, string operators, string methods, the `if` statement, and the `return` statement. It's fine if you choose to use comparison operators as well. *Partial credit* is possible.

   Comment:

9. _____ Your `pigWord` function calls the `findVowel` function. Your `pigWord` function makes good use of Python's string-handling features (Table 3.1 and Table 3.2 of the textbook), avoiding excess complexity. For example, you should not import modules or use technology other than string operators and methods. You do not need a for loop. *Partial credit* is possible.

10. _____ You submitted exactly one `.py` file called `project3.py`
Your Python file `project3.py` runs without error on IDLE 3 using one of the Olin 326 machines.
Give your file this exact name and extension. (Note the lower case letters and the lack of white spaces).
Partial credit is possible.

    Comment:

11. _____ Your `isLegal` function *returns* a *boolean* answer. You will get credit even if the function returns incorrect answers for some sample inputs. No point if the `return` statement is not used, or if a value returned is not boolean for certain sample inputs. **No partial credit.**

    Comment:

12. _____ Your `isLegal` function *returns* a *boolean* answer. In addition, your `isLegal` function correctly checks if the weapon chosen by a player is either "rock", "paper", or "scissors". **No partial credit.**

    Comment:

13. _____ Your `beats` function returns a *boolean* answer. You will get credit even if the function returns incorrect answers for some test inputs. No point if the `return` command is not used, or if the value returned is not boolean. **No partial credit.**

    Comment:

14. _____ Your `beats` function returns a boolean answer. In addition, your `beats` function correctly identifies all sample cases in which the first player would win (or lose) in a round of rock-paper-scissors. **No partial credit.**

    Comment:

15. _____ Your `unPigWord` function correctly locates the hyphen. Your `unPigWord` function correctly extracts the substring preceding the hyphen. Your `unPigWord` function correctly extracts the substring between the hyphen and the ending. *Partial credit* is possible.

16. _____ Your `unPigWord` function *returns* correct values for 100% of all sample inputs. **No partial credit.**

    Comment:

17. _____ Your `findVowel` function returns the specified value when given *a string with no vowel.* For example, your function returns the correct values for input strings such as `'my'`, `'gym'`, `'myth'`, `'try'`, `'rhythm'`. You will get the full one point even if your function does not work for strings containing vowels. **No partial credit.**

    Comment:

18. _____ Your `findVowel` function correctly returns the correct integer for 100% of all sample inputs. **No partial credit.**

19. _____ Your `pigWord` function finds the dividing position within the given string. Your `pigWord` function correctly extracts the substring prior to the dividing position. Your `pigWord` function correctly extracts the substring starting at the dividing position. *Partial credit* is possible.

    Comment:

20. _____ Your `pigWord` function *returns* correct values for 100% of all sample inputs. **No partial credit.**

    Comment:

_____ Total points