Lab 6 Problems

Problem 1 In this problem, you will be implementing a stack using a linked list as the underlying data structure.

Step 1: Download the file `lab6_starter.py` from the class website (use the Labs link).

Step 2: Complete the code for the `Stack` class

Step 3: Complete the steps in the `main()` function. When asked to draw a diagram, draw it below:

Problem 2 In the Assignment 6 long problem, you will be reading a file that consists of the titles of news articles. When processing each line of the input file, one of your tasks will be to remove all punctuation from the title in the input line.

Step 1: Go to the class website and pull up the Assignment 6 specification. Search for **Processing titles** and read that section.

Step 2: Use the suggestion given there to complete the function `remove_punc(title)` that takes a string `title` and returns a *new* string with all punctuation removed. For example, `remove_punc("It's a dog's life, full of treats!")` returns the string

```
'It s a dog s life  full of treats '
```

Complete the function `remove_punc()` in `lab6_starter.py` and call the function from main.

Problem 3 In Assignment 6, you will be using your sort() method from Assignment 5 to sort a linked list. The nodes will have three attributes, `_count`, `_word` and `_next`.

Step 1: Go to the **Expected Behavior** section and look at the output in item 4.

Step 2: Are the nodes sorted by the `_count` or the `_word` attribute?

Step 3: Should the output then also be sorted alphabetically by the words?

Step 4: We know that comparison operators work on numbers, but you may not know that they also work on strings. For example, for the comparison below,

```
"bye" < "hello"
```

Python will compare two the operand strings character by character and return `True` if the first operand is lexicographically less than the second operand (that is, if `"bye"` would appear in a human-dictionary before `"hello"`.)

In `lab6_starter.py`, fill out the function `string_comp()` as shown below and then call `string_comp()` in `main()` to verify the results of comparing strings:

```
def string_comp():
    print("apple" < "banana")
    print("banana" < "bananas")
    print("hello" == "hello")
    print("bye" < "hello")
    print("hello" < "bye")
```

Problem 4 The data in the lines in the input file in Assignment 6 are separated by commas. You might initially think that you can simply read in a line and split on the commas. However, a title in an input line might also contain a comma.

Step 1: Go to the **Expected Behavior** section. What information does the spec give you about reading the file?

Step 2: What are the two modules that you are allowed to import in Assignment 6?