Homework 1

Due Sept 7, 2020, 4PM EST

* Preliminaries. You'll need to list your first and last name with each homework assignment. Fill in your name in the box below. * Do your work in this notebook. There are some cells reserved to hold your answers. * Other cells hold "Unit Tests" - small bits of code to test your work. Run those to be sure you are on track It will be easier to provide Unit Tests for all problems as we learn more * When you are ready, clear all results (Cell/All Output/Clear) and then run all cells (Cell/Run All). This will run the cells in order, and can help you find any errors you might have missed. * If you find a problem you cannot fix, run the cells below it by hand. * Save your version of the notebook and save it again as a pdf file and submit both files to Canvas. Be sure that we can see the results of each cell in the PDF.

In [43]:
```
# Fill in your name

first_name = 'Shariq'
last_name = 'Jamil'

assert len(first_name) != 0, "First name is blank"
assert len(last_name) != 0, "Last name is blank"
```

Problem 1 (3 points)

In the cell below we define two strings

```
a= "+"

b = "-"
```

Using a and b, write a series of python statements to create the 45-character string below, and assign it to the variable s, using as few operations as possible:

```
"+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+"
```

In [44]:
```
# Do not change this cell

a = "+"
b = "-"
```

Put your code in the cell below.

In [45]:
```
s = a + b

# The number of times the loop will run
num_of_times = 21
# Keeps track of where the loop is at
counter = 0

# Perform operation 21 times
while counter < num_of_times:
    # Append '+-' to variable s
    s += (a + b)
    # Increment counter
    counter += 1

# Append a final '+' to the string
s += a

print(s, len(s))
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ 45
```

Unit Test for Problem 1

Run the cell below to test your value of s.

In [46]:
```
# This cell will test your string s

assert s[:2] == '+-', "String must start with +-"
assert s[2:] == s[:-2], "Pattern must repeat"

assert len(s) == 45, "Wrong length"

print('Success!')
```

```
Success!
```

Problem 2 (4 points)

In the cell below we define a string called *quotation*.

Write Python expressions to compute each of the values below, and print each in turn.

- The length of the string *quotation*.
- The index of the first instance of the word 'country'. The index of 'Ask' is 0.
- The index of the last instance of the word 'country'.
- The number of times 'you' appears in *quotation*, using the string method count().

Put your code in the cell below.

In [47]:
```
quotation = "Ask not what your country can do for you - ask what you can do for you
r country."

# Place 4 expressions below
print(len(quotation))                    # Length
print(quotation.index('country'))        # Index of first instance of 'country'
print(quotation.rfind('country'))        # Index of last instance of 'country'
print(quotation.count('you'))            # Number of times 'you' appears in quotatio
n
```

```
80
18
72
4
```

Problem 3 (3 points)

Write a Python fragment to create a string holding the text below.

```
Then he said "I don't know."
```

Put your code in the cell below.

In [48]:
```
# Write an expression to initialize the variable s.

# Used backslash to escape single quote in the middle of the double quoted substrin
g
s = 'Then he said "I don\'t know."'
```

Unit Test for Problem 3

Run the cell below to test your value of s.

In [49]:
```
# print your string
print(len(s), s)

# Check that it is the right length
assert 28 == len(s), "Wong length"

# Check the contents
assert 22 == s.find('know'), "'Know' should be at index 22"
assert 22 == s.rfind('know'), "'Know' should be at index 22"

print('Success!')
```

```
28 Then he said "I don't know."
Success!
```

Problem 4 (2 points)

Import the datetime library and write a Python statement that computes and prints the next year.

When we run it next week, it should print 2021.

Put your code in the cell below.

In [50]:
```
import datetime

# Increment the current year by 1
next_year = datetime.date.today().year + 1

# Print the next year
print(next_year)
```

```
2021
```

Problem 5 (3 points)

Write a Python expression to exchange the front and back halves of a word.

Your fragment should be able to change 'look' to 'oklo' and 'frost' to 'ostfr'.

It should change 'Problem' to 'lemProb'

Put your code in the cell below.

In [51]:
```
word = 'frost'

# find middle of the given word. Using floor division for odd numbered words
middle_of_word = len(word) // 2

# set first half of the word
first_half = word[0:middle_of_word]

# set second half of the word
second_half = word[middle_of_word:len(word)]

flip = second_half+first_half      # Place your expression here.
```

Unit Test for Problem 5

Run the cell below to test your value of flip. You expression should work flip the front and rear of any word.

In [52]:
```
assert flip == 'ostfr', "Didn't flip 'frost' properly"

print('Success!')
```

```
Success!
```

Documentation (5 points)

Use comments and meaningful variable names to make your Python clear

Post Mortem

How long did it take you to solve this problem set? Did anything confuse you or cause difficulty?

In [53]:
```
# Your thoughts...
# It took me about thirty minutes. I was not sure specifically to use to solve each
problem so hopefully I am not working too far ahead.
```