CSCA08 Exercise 4

Due: October 22, 2017. 5:00pm
Pre-Run: October 20, 2017. 5:00pm

The functions in this exercise are designed to be solvable in a number of ways. Just getting them working shouldn't be much of a challenge, but exploring different ways of solving the same problem will be fun. Try to solve each function using loops and then again using slicing (you can submit whichever version you wish). As always, all of your functions should be in appropriately named files (`ex4.py` and `ex4.test`), and you should not use `print`, `input` or `import`.

insert

Write a function called `insert` that takes 3 parameters, `listA`, `listB`[1] and an index, then returns a copy of `listA` with the elements of `listB` inserted at the index. Your code should work with either strings or lists e.g.,

```
>>> insert([1, 2, 3], ['a', 'b', 'c'], 2)
[1, 2, 'a', 'b', 'c', 3]
>>> insert("123","abc",2)
'12abc3'
```

up_to_first

Write a function called `up_to_first` that takes two parameters, a list (or string) and an object, and returns a copy of the list up to (but not including) the first occurrence of that object, or all of the elements if that object is not in the list. e.g.,

```
>>> up_to_first([1, 2, 3, 4], 3)
[1, 2]
>>> up_to_first([1, 2, 3, 4], 9)
[1, 2, 3, 4]
>>> up_to_first('abcdef', 'd')
'abc'
```

cut_list

Write a function called `cut_list` that "cuts" a list. That is, given a list and an index, returns a copy of the list, but with the items before and after the index swapped. As always, your code should also work with strings. e.g.,

```
>>> cut_list([0,1,2,3,4,5,6,7,8,9], 3)
[4, 5, 6, 7, 8, 9, 3, 0, 1, 2]
>>> cut_list("ABCDEFGX1234",7)
'1234XABCDEFG'
```

[1] You should probably come up with better names than these