CSCA08 Exercise 6

Due: November 3, 2013. 5:00pm

This week, we're working with mutability and the memory model. You will be writing two functions that do **almost** the same thing, but deal with input/output in very different ways. As always your code should be written in a single file `ex6.py`, and you should not be using any `print/input/import` statements.

Changing a Copy of a List

Write a function called `copy_me` that takes as input a list, and returns a copy of the list with the following changes:

- Strings have all their letters converted to upper-case
- Integers and floats have their value increased by 1
- booleans are negated (False becomes True, True becomes False)
- Lists are replaced with the word "List"

The function should leave the original input list unchanged (i.e., do not mutate the list)

Mutating a List

Write a function called `mutate_me` that takes as input a list, returns None, and changes the input list in the following ways:

- Strings have all their letters converted to upper-case
- Integers and floats have their value increased by 1
- booleans are negated (False becomes True, True becomes False)
- Lists are replaced with the word "List"

So we're performing the same task, but this time we're changing the list itself instead of making a copy to return.

Memory Model Work

On a piece of paper, trace the following code in the memory model:

```
def my_func(input_list):
    input_list[0][2] = 99


def my_func2(input_list):
    input_list[0] = [1, 2, 3]
    my_func(input_list)

my_list = [1, [3, 2, 1], [1, 2]]
my_func2(my_list)
```

If you get stuck, you can try it in the visualizer, but try your best to do the entire thing by hand. (This can count towards your hour of memory model practice).

What to hand in

I hear you saying: “Brian, we know how to hand in the first two functions on MarkUs, but what about the memory model drawing? Surely you can’t auto mark that, and I doubt you’d make us scan our drawings and submit them, would you?”. You’re correct. We can’t mark the memory model work in any efficient way. However, we really don’t need to, because you can check your answers yourself in the visualizer. Remember that the primary goal of all of this is to learn/practice. I’m just going to trust you that you’ve done it, and if you decide not to bother, then you’re just depriving yourself of some good midterm practice.