

# CS 161 Lab Activity: Functions

## Part One: Defining and Calling a Function

Try each of these sequences in the interactive environment and study the results. (You may need to type some in a small program file and run it to get satisfactory results. If any results seem strange, study the code and figure out why it works the way it does.

1)

```
>>> def funny ( ):
...     print "yes"
...
>>> funny ()    # calls the function
>>> funny      # shows its address
```

2)

```
>>> def sing ( ):
...     print "99 bottles of beer on the wall, "
...     print "99 bottles of beer,"
...     print "take one down, and pass it around,"
...     print "98 bottles of beer on the wall."
...
>>> sing ( )
>>> sing ( 12 )    # doesn't work- why not?
```

3)

```
>>> def sing (howMany):
...     print howMany, "bottles of beer on the wall, "
...     print howMany, "bottles of beer,"
...     print "take one down, and pass it around,"
...     print howMany - 1, "bottles of beer on the wall."
...
>>> sing (99)
>>> x = 52
>>> sing (x)
>>> sing ("Fred")    # doesn't work- why not?
>>> for bottleNumber in range (99, 70, -1):
...     sing (bottleNumber)
...
...

```

Continues on next page...

## Part Two: A Program with Functions

You will need to create a program file for this one, and then execute it from the command line. **Observe** what happens when you attempt to run the program at each stage.

1) First, create a file with this code in it:

```
def doSomething ( x ):
    x = x * 10

def main ( ):
    x = 5
    doSomething ( x )
    print "x is ", x
main ( )
```

2) Next, add this statement to the end of function doSomething: **return x** and then run the program again.

3) Next try changing the variable names in the function doSomething to y and run the program again.

4) Next change the function call to **x = doSomething( x )** and run the program again.

5) Now try replacing the call with **x = doSomething( 14 )** and run it again.

Here is a short program that will let you try different interest rates and investment amounts to see what the future value of the investment at the interest rate would be. As written, it doesn't quite work. Fix it.

```
def computeValue (principal, interest, time):
    for year in range (time):
        principal = principal * (1 + rate)
    return principal

def main ( ):
    prompt = "enter an amount to invest, or zero to quit"
    amount = input (prompt)

    while amount != 0:
        rate =input ("enter the interest rate (e.g.,.05) ")
        futureValue = computeValue (rate, amount)
        print "The value will be ", futureValue
        amount = input (prompt)

main()
```

(If you have trouble finding the problem, read the error message, and look at the function header *and* the call).