University of Pittsburgh

CS 0008 Spring 2015 - Information about Exam 1

- Exam 1 will focus on:
  - the material and code covered in lecture through 2/03
  - the reading assignments listed on the schedule through 2/03
  - Lab 2 (though Lab 3 from the Monday before will also be useful)
  - Assignment 1
- The following comment will appear on the front of the exam:

  Unless you think there is an error, I can't answer any questions. Otherwise, it gets too noisy, and some people would get more information than others.

  Please just follow the instructions, and use your best judgment.
- The exam is closed everything except you and your pencil (ok, you may bring an eraser/sharpener).
- Study actively. Don't just pass your eyes over the text and materials from lecture. Run the code. Try your own commands in the Python shell. If you are confused, trace code on paper then run the code to check yourself. You can use the debugger or pythontutor.org to see the values that variables are assigned as the program executes.

Below are the types of questions, with examples of each type, that will be on the exam.

The purpose of this is to give you an idea of what to expect. Some questions here are directly out of the lectures or assignment. The actual questions will **not be**, but they will be related to something covered in lecture, the text, labs, or Assignment 1.

These examples are **just samples**. The actual exam will be longer, and will cover additional programming features and topics. Also, the exam will include easy, medium, and hard questions.

Short answer question.

- What does this condition evaluate to? (True or False) and (False or False)
- What value does this expression evaluate to? 1 + 3 * 7
- What value does this expression evaluate to? (1 + 3) * 7

Fixing an error.

- Suppose you want to print 'yes' if the value of variable x is between 0 and 10 (inclusive). The following code is not correct. Please fix it.

```
if x >= 0 and <= 10:
    print('yes')
```

Tracing code. Hint: The question on the exam will ask for some specific variable values, as shown in this question. But, use the extra whitespace on the side to trace the variables fully as in Lab 2, so you are sure what is going on.

- Consider the code below. This is on the hard side.

```
# Reminder:  range(0,3) returns a sequence from 0 through 2.
def mystery(x):
    y = 0
    z = x
    for i in range(0,3):
       print("i =", i)
       if i  > 1:
           y = y + x
       z = z + 1
    return y

y = 5
x = mystery(y)
print("Result:",x)
```

Trace the variable values during execution of this program. Specifically, for each underline in the table below, fill in the value of that variable after the line has executed, or write ‘NR’ (for ‘not reached’) if that line was not executed.

```
                                   during iteration
                              1          2        3

y = y + x   mystery.y     ------  ------  ------

z = z + 1   mystery.z     ------  ------  ------
```

Giving the output of some code.

- Give the output of the following program.

```
def fun1(x):
   print ("!!",x * 2)
   print (fun2(x * 2))
   return x * 2

def fun2(y):
   print("inside fun2, y is", y)
   return y - 1

def main():
   print ("Splat")
   print ("Calling fun1 now")
   print ("fun1(5) is",fun1(5))

main()
```

Writing a function.

- Write the function specified below.

```
def num_vowels(s):
    '''Return the number of vowels in string s.
    The letter ‘‘y’’ is not treated as a vowel.'''
```

Calling functions to do something.

- Assume the following functions have already been defined:

```
def remove_rightmost_digit(num):
    '''num: an int.
       return num with the rightmost digit removed.
       E.g. remove_rightmost_digit(5432) returns 543
       If num has only one digit, return 0'''


def get_rightmost_digit(num):
    '''num: an int.
```

```
       return the rightmost digit of num.
       E.g. remove_rightmost_digit(5432) returns 2 ’’’
```

Assume that num has been assigned an int that is at least 4 digits long. Write code to extract and print the 3rd digit from the right. For example, if num is 45678, your code should print 6.