

# University of Pittsburgh

## CS 0007 Fall 2009 Test 1 - Sample

There are four questions for a total of 100 points.

Some Python function/method descriptions are given to you.

Otherwise, no other aides or notes are permitted.

We can't help you answer the 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.

Good luck!

1. (25 points) Consider the code below.

```
def mystery(x):
    y = 0
    z = x
    for i in range(0,4):
        if i * 10 > y:
            y = y + x
            z = z + 5
    return y
```

```
if __name__ == '__main__':
    y = 11
    x = mystery(y)
    print x
```

Trace the variable values during execution of this program. Specifically, for each underline in the table below, fill in the value of the variable specified after the line has executed, or write 'NR' (for 'not reached') if that line was not executed. Note: I will use a longer program on the real test, so there is more chance for partial credit. The longer program will include more interesting boolean expressions, and it will definitely have a for loop and if statements. Also, it will test your knowledge of scope more than this one does.

def mystery(x):	show values after the line executes			
	during iteration			
	1	2	3	4
y = y + x	y: __NR__	__11__	__22__	__33__
z = z + 5	z: __16__	__21__	__26__	__31__

2. (25 points) Complete the following function according to its docstring.

```
def modify_colors(pic):
    '''Modify the picture 'pic' so that any white pixels are replaced
    with blue and any black pixels are replaced with white.  Pixels of
    any other color remain unchanged'''

    for pix in pic:
        if media.get_color(pix) == media.white:
            media.set_color(pix,media.blue)
        elif media.get_color(pix) == media.black:
            media.set_color(pix,media.white)
```

3. (20 points)

Complete the following function according to its docstring description. Note: this one is too easy. I was showing that there may be a question closely related to a lab question.

```
def remove_rightmost_2_digits(num):
    '''Return int num, but with the rightmost 2 digits removed.
       Example: remove_rightmost_digit(4235) returns 42
       You can assume that the num has at least 3 digits'''

    return num / 10 / 10
```

4. (30 points) For this question, write a program that prompts the user for a picture file (using `media.choose_file`), and then prompts the user for a color, either 'red', 'green', or 'blue' (using `raw_input`). Your program should load the picture, and then, for the color component selected, for each pixel in the picture, it should set that color component to the absolute value of the difference of the other two color components of that pixel. At the end, your program should display the changed picture (using `media.show`).

Your program should define and call exactly one function, `diff_color`. Complete that function according to the given docstring. Then, below it, write the main program, which calls this function.

```
if __name__ == '__main__':

def diff_color(pic,color):
    '''Given a picture 'pic' and a string color ('red', 'green', or
    'blue'), for each pixel in 'pic', set that color component to the
    absolute value of the difference of the other two color components
    of that pixel.'''

    for pixel in pic:
        if color == "red":
            g = media.get_green(pixel)
            b = media.get_blue(pixel)
            media.set_red(pixel,abs(g - b))
        elif color == "blue":
            r = media.get_red(pixel)
            g = media.get_green(pixel)
            media.set_blue(pixel,abs(r - g))
        else:
            r = media.get_red(pixel)
            b = media.get_blue(pixel)
            media.set_green(pixel,abs(r - b))

if __name__ == '__main__':

    fn = media.choose_file()
    pic = media.load_picture(fn)
    c = raw_input('Enter red, green, or blue ')
    diff_color(pic,c)
    media.show(pic)
```

Short Python function/method descriptions:

`__builtins__`

`abs(number)` -> number  
Return the absolute value of the argument.

`raw_input([prompt])` -> string  
Read a string from standard input.

In Media:

`choose_file()`  
Prompt user to pick a file. Return the path to that file.  
Change the current working directory to the directory  
where the file chosen by the user is

`create_picture(w, h, col=Color(255, 255, 255))`  
Return a Picture w pixels wide and h pixels high.  
Default Color col is white.

`get_blue(pix)`  
Return the blue value of Pixel pix.

`get_red(pix)`  
Return the red value of Pixel pix.

`get_green(pix)`  
Return the green value of Pixel pix.

`get_color(pix)`  
Return the Color object with Pixel pix's RGB values.

`get_pixel(pic, x, y)`  
Return the Pixel object at the coordinates (x, y) in Picture pic.

`load_picture(filename)`  
Return a Picture object from filename filename.

`set_blue(pix, b)`  
Set the blue value of Pixel pix to b.

`set_green(pix, g)`  
Set the green value of Pixel `pix` to `g`.

`set_red(pix, r)`  
Set the red value of Pixel `pix` to `r`.

`set_color(pix, col)`  
Set the RGB values of Pixel `pix` to those of Color `col`.

`show(pic)`  
Display Picture `pic` in separate window.

Colors:

`media.black`

`media.white`

`media.blue`