CS 111 - Winter 2020
Quiz 3 Practice Questions

The quiz will be worth 30 points, and each point should correspond roughly to one minute of your time. This list of practice questions is not necessarily representative of the length of the quiz – it is hopefully longer to give you more practice, but there might be more questions in a given category on the actual quiz.

True/False questions (approximately 2 points)

1) The CPU is the "brain" of the computer.

2) All information that a computer is currently working on is stored in main memory.

3) The process of associating a file with an object in a program is called "reading" the file.

4) The `main` function is at the bottom of the functional structure chart (the chart we drew while designing a coin-flipping game to describe how functions connect).

5) Unit testing is the process of trying out a single component of a larger program in isolation.

Multiple-choice questions (approximately 3 points)

1) What color is `color_rgb(0, 255, 255)`?
(a) yellow **(b)** cyan **(c)** magenta **(d)** orange

2) Which of the following is *not* a file-reading method in Python?
(a) `read` **(b)** `readline` **(c)** `readall` **(d)** `readlines`

3) Which expression is true approximately 20% of the time?
(a) `random() < 0.2`
(b) `random() <= 0.2`
(c) `random() <= 20`
(d) `random() < 20`

4) In top-down design, the subcomponents of the design are
(a) objects **(b)** loops **(c)** functions **(d)** programs

Code prediction (approximately 9 points)

1) Predict what the image drawn by the following code would look like, by describing the color(s) at a few points or ranges of points.

```
image = Image(Point(150,150), 256, 256)

i = -1
for row in range(image.getHeight()):
    for col in range(image.getWidth()):
        if col % 8 == 0:
            i = (i+1) % 256
        val = min(i, 255)
        color = color_rgb(0, val, val)
        image.setPixel(col, row, color)
```

(a) $(0, 0)$

(b) $(149, 149)$

(c) $(col, 8)$, where *col* ranges from 0 to 149

(c) $(0, row)$, where *row* ranges from 0 to 7

2) Assume you have a file containing the following text:

```
100,200,10,10,180
50,300,250,200,10
200,100,80,0,0
```

What does the following code do, assuming this file is named “myfile.txt”? Describe the result of parsing and using each line of the file.

```
image = Image(Point(200, 200), 400, 400)

f = open("myfile.txt", ’r’)

for line in f:
    s = line.strip().split(’,’)
    v0, v1 = int(s[0]), int(s[1])
    for i in range (v0-5, v0+6):
        for j in range(v1-5, v1+6):
            image.setPixel(i, j, color_rgb(int(s[2]), int(s[3]), int(s[4])))

f.close()
```

3) What are three different likely outputs for the following code?

```
count = 0
n = 1000
for i in range(n):
    val = random.randint(0,9)
    if val < 3:
        count += 1
print("The final count was", count, "out of", n)
```

Rewriting code (approximately 6 points)

1) Rewrite the three `filter*Vals` functions to be a single function that take another function as an additional parameter. You should also make the necessary changes to `main` to call your new function. The overall program should have the same behavior as the original. (Note that this is more points than passing functions as parameters will get on the actual quiz, but should give you good practice.)

```
def isEven(val):
    return val % 2 == 0

def isOdd(val):
    return val % 2 == 1

def isNegative(val):
    return val < 0

def filterEvenVals(mylist):
    res = []
    for val in mylist:
        if isEven(val):
            res.append(val)
    return res

def filterOddVals(mylist):
    res = []
    for val in mylist:
        if isOdd(val):
            res.append(val)
    return res

def filterNegativeVals(mylist):
    res = []
    for val in mylist:
        if isNegative(val):
            res.append(val)
    return res

def main():
    myvals = [1,5,4,1,-3,0,-2,-3,10]
    evenVals = filterEvenVals(myvals)
    oddVals = filterOddVals(myvals)
    negVals = filterNegativeVals(myvals)

    print("Even values:", evenVals)
    print("Odd values:", oddVals)
    print("Negative values:", negVals)

if __name__ == "__main__":
    main()
```

Your new function:

```
def filterVals(mylist, func):
    # TODO: convert three functions into one that takes func as a new parameter
```

```
def main():
    # TODO: rewrite main
```

Fixing bugs (approximately 4 points)

1) The following program is meant to read in a text file like the following:

```
name,weight,months
Therese,6.5,186
Sadie,56,144
Hobbes,11.8,94
Lulu,9,95
Colin,24.2,17
```

The program should perform two functions. It should output the text to a new file “outfile.csv” with a new column indicating the years in age of the pet/baby, and it should generate a bar chart of the weight data. Unfortunately, there are six bugs. For each bug, identify it and fix the code. Notes:

- The Python keyword in isn’t just for for loops. If used in an expression within an if/elif statement or while loop, it results in a Boolean. For strings, it checks whether the left-side string is a substring of the right-side string. For lists, it checks whether the left-side operand is an element within the right-side list.
- The function matplotlib.pyplot.bar takes in two lists, one indicating the x-axis labels, and the other indicating the values on the y-axis. Typically, these should be numbers.

```
from matplotlib import pyplot as plt

def main():
    fin = open("myfile.csv", ’r’)
    fout = open("outfile.csv", ’r’)

    nameList = []
    weightList = []

     for line in fin.readline():
         if line.strip() == "":
             continue
         elif "name" in line: # check if "name" is a substring of the line contents
             outline = line.strip() + ",years"
             print(outline, file=fout)
         else:
             s = line.strip().split(’,’)
             name, weight, months = s
             nameList.append(name)
             weightList.append(weight)
             years = months // 12
             vals = [name, str(weight), str(months), str(years)]
             outline = ",".join(vals)
             print(outline, file=fin)
```

```

        fin.close()

        plt.figure()
        plt.bar(nameList,weightList)
        plt.show()

    if __name__ == "__main__":
        main()
```

Writing code (approximately 6 points)

1) A Monte Carlo simulation (similar to flipping coins or rolling dice) can be used to estimate the value of pi. Suppose you have a round dartboard that just fits inside a square cork board. If you throw darts randomly (but always hit the square at least), the proportion that hit the dartboard versus those that hit the square cork board will be determined by the relative area of the dartboard and the cork board.

If n is the total number of darts randomly thrown, and d is the number that hit the dartboard, it can be shown that $\pi \approx 4(d/n)$.

Write a program that asks the user for n (the "number of darts") as input, and then performs a simulation to estimation π.

Hints:

- You can use `2*random.random() - 1` to generate the x and y coordinates of a random point inside a 2x2 square centered at $(0, 0)$. The point lies inside the circle if $x^2 + y^2 \leq 1$.
- It is wise to use top-down design on such a problem, like we did with simulating the coin-flipping game.

Note that this problem is *much* longer than one you'd see on the quiz, but it'd be valuable to practice solving it, because it hits on many of the key points.