Mock Final Exam
CSci 127: Introduction to Computer Science
Hunter College, City University of New York

15 May 2018

Answer Key:

1. (a) What will the following Python code print:

    i.
    ```python
    s = "Ada:)Lovelace:)Grace:)Hopper"
    a = s[0:3]
    print(a.upper())
    ```

    Answer Key:
    ```
    ADA
    ```

    ii.
    ```python
    names = s.split(":)")
    print(names[-1])
    ```

    Answer Key:
    ```
    Babbage
    ```

    iii.
    ```python
    b,c,d = names[1],names[2],names[3]
    print(b,d)
    ```

    Answer Key:
    ```
    Lovelace Babbage
    ```

    iv.
    ```python
    print(b[-1]+"n"+d[-2]+"ine")
    print('Put_line: ("', a.lower(),'")')
    ```

0

Answer Key:
```
engine
Put_line: (" ada ")
```

(b) Consider the following shell commands:

```
$ ls -l *z*
-rw-r--r--@ 1 stjohn  staff      5308 Mar 21 14:38 quizzes.html
-rw-r--r--  1 stjohn  staff     54013 Mar 20 18:57 zoneDist.csv
-rw-r--r--@ 1 stjohn  staff      1519 Mar 22 15:14 zoneMap.py
-rw-r--r--  1 stjohn  staff  16455174 Mar 20 19:02 zoning2.html
-rw-r--r--  1 stjohn  staff  17343896 Mar 20 18:58 zoningIDS.json
```

  i. What is the output for:
```
$ ls -l *z* | grep ".html"
```

Answer Key:
```
-rw-r--r--@ 1 stjohn  staff      5308 Mar 21 14:38 quizzes.html
-rw-r--r--  1 stjohn  staff  16455174 Mar 20 19:02 zoning2.html
```

  ii. What is the output for:
```
$ ls -l *z* | grep ".html" | wc -l
```

Answer Key:
```
2
```

2. (a) Fill in the missing code below:

```
#Demonstrates colors, using turtles
import turtle
tess = turtle.Turtle()
#Set color of tess to blue:
```

```
```

```
#Set color of tess to maximum red, maximum blue, and no green:
```

Answer Key:

```
tess.color(255,255,0)
```

```
#Set color of tess using hexcodes: red, green, and blue all equal to "A0":
```

Answer Key:

```
tess.color("#A0A0A0")
```

(b) Write the Python code for the following algorithm:

```
Ask user for input, and store in the string, binString.
Set decNum = 0.
For each c in binString,
    Set n to be int(c)
    Double decNum and add n to it (decNum = 2 * decNum + n)
Print decNum
```

Answer Key:

```
#binary to decimal
binString = input('Enter num: ')
decNum =0
for c in binString:
     n = int(c)
      decNum = decNum * 2 + n
print(decNum)
```

3. (a) What is the value (True/False):

```
    in1 = False
 i. in2 = True
    out = in1 and in2
```

Answer Key:

```
out = False
```

```
     in1 = False
 ii. in2 = True
     out = not in1 or (in2 and not in1)
```
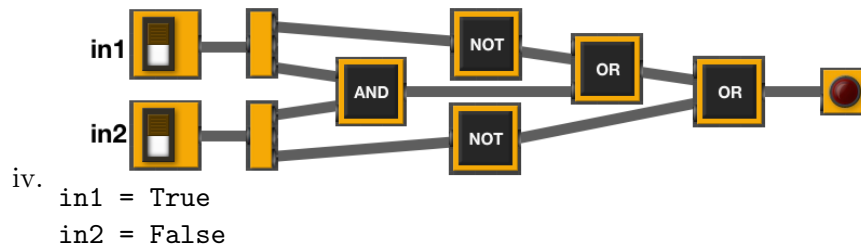
Answer Key:

```
out = True
```

```
      in1 = False
      in2 = False or not in1
 iii.
      in3 = in1 and in2
      out = in1 and not in3
```
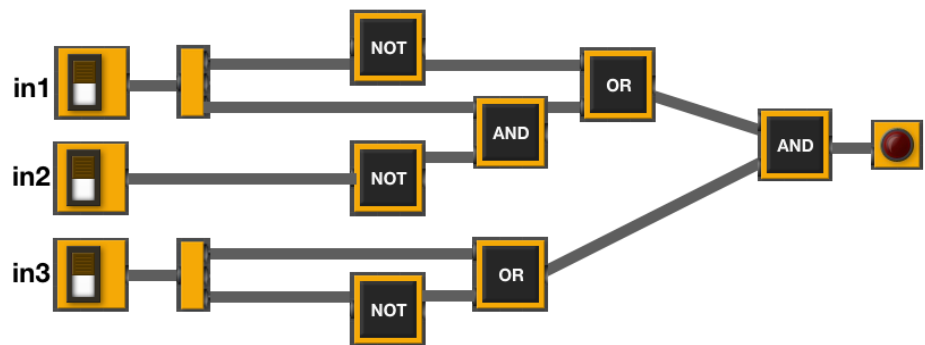
Answer Key:

```
out = False
```

iv.


```
in1 = True
in2 = False
```

Answer Key:

```
out = True
```

(b) Design a circuit that implements the logical expression:

```
((not in1) and (in1 or not in2)) and (in3 or not in3)
```


Answer Key:

4. (a) Draw the output for the function calls:

```
import turtle
tess = turtle.Turtle()
tess.shape("turtle")

def ramble(t,length):
    if length < 10:
        t.stamp()
    else:
        t.forward(length)
        t.left(90)
        ramble(t,length-10)
```

i. `ramble(tess,0)`

Answer   **Key:**

ii. `ramble(tess,50)`

Answer   **Key:**

(b) What is returned when the function is invoked on the inputs below:

```
def gee(a,b):
    while a != b:
        if a > b:
            a = a - b
        else:
```

```
                    b = b - a
             return a
```

   i. `gee(1,1)`

      Answer Key: 1

   ii. `gee(2,3)`

      Answer Key: 1

   iii. `gee(2,4)`

      Answer Key: 2

   iv. `gee(16,24)`

      Answer Key: 8

5. Write a **complete Python program** that uses `folium` to make a map of New York City. Your map should be centered at $(40.75, -74.125)$ and include a marker for the main campus of Hunter College. The HTML file your program creates should be called: `nycMap.html`. **Answer Key:**

```
#Make a map centered in NYC and with popup marker for Hunter

import folium

mapCUNY = folium.Map(location=[40.75, -74.125], zoom_start=10)
folium.Marker(location = [40.768731, -73.964915], popup = "Hunter College").add_to(mapCUNY
mapCUNY.save(outfile='nycMap.html')
```

6. Using `matplotlib.pyplot` and `numpy`, write a **complete Python program** that reads in an array (grid) of elevations, `elevations.txt`. Your program should create an image where for each element of the array, the corresponding pixel is colored in the final image is:

   - colored blue if the elevation is 0 or less,
   - black if the elevation is positive and divisible by 10, and
   - gray otherwise.

   Your resulting image should be stored in a file, `topoMap.png`.

   Answer Key:

```
# Takes elevation data of NYC and displays a contour map
```

```
#Import the libraries for arrays and displaying images:
import numpy as np
import matplotlib.pyplot as plt

#Read in the data to an array, called elevations:
elevations = np.loadtxt('elevationsNYC.txt')

#Take the shape (dimensions) of the elevations
#  and add another dimension to hold the 3 color channels:
mapShape = elevations.shape + (3,)

#Create a blank image that's all zeros:
topoMap = np.zeros(mapShape)

for row in range(mapShape[0]):
    for col in range(mapShape[1]):
        if elevations[row,col] <= 0:
            #Below sea level
            topoMap[row,col,2] = 0.5
        elif elevations[row,col] % 10 == 0:
            topoMap[row,col,0] = 0.0
            topoMap[row,col,1] = 0.0
            topoMap[row,col,2] = 0.0
        else:
            topoMap[row,col,0] = 0.3
            topoMap[row,col,1] = 0.3
            topoMap[row,col,2] = 0.3

#Save the image:
plt.imsave('topoMap.png',topoMap)
```

7. Fill in the following functions that are part of a program that analyzes NYC Urban Forest of street trees (from NYC OpenData):

   - `getData()`: asks the user for the name of the CSV file and returns a DataFrame of the contents.
   - `totalTrees()`: returns the number of trees (length) in the DataFrame, and
   - `biggestDiameter()`: returns the largest diameter (`tree_dbh`) in the DataFrame.

   Answer Key:

```
import pandas as pd
def getData():
    """
```

```
        Asks the user for the name of the CSV and
        Returns a dataframe of the contents.
        """
        fileName = input('Enter file name: ')
        df = pd.read_csv(fileName)
        return(df)
```

```
    ┌─────────────────────────────────────────────────────────────┐
    │                                                               │
    │                                                               │
    │                                                               │
    │                                                               │
    │                                                               │
    └─────────────────────────────────────────────────────────────┘
```

```
    def totalTrees(df):
        """
        Takes a DataFrame as input.
        Returns the length of the DataFrame.
        """
        length = len(df)
        return(length)
```

```
    ┌─────────────────────────────────────────────────────────────┐
    │                                                               │
    │                                                               │
    │                                                               │
    │                                                               │
    │                                                               │
    └─────────────────────────────────────────────────────────────┘
```

```
    def biggestDiameter(df):
        """
        Takes a DataFrame as input and
        Returns the maximum value in
        the column, tree_dbh.
        """
        M = df['tree_dbh'].max()
        return(M)
```

8. (a) What is the output for a run of this MIPS program:

```
        # Store 'Help!!' at the top of the stack
        ADDI $sp, $sp, -7
        ADDI $t0, $zero, 72 # H
        SB $t0, 0($sp)
        ADDI $t0, $zero, 101 # e
        SB $t0, 1($sp)
        ADDI $t0, $zero, 108 # l
        SB $t0, 2($sp)
        ADDI $t0, $zero, 112 # p
        SB $t0, 3($sp)
        ADDI $t0, $zero, 33 # !
        SB $t0, 4($sp)
        ADDI $t0, $zero, 33 # !
        SB $t0, 5($sp)
        ADDI $t0, $zero, 0 # (null)
        SB $t0, 6($sp)
        ADDI $v0, $zero, 4 # 4 is for print string
        ADDI $a0, $sp, 0
        syscall  # print to the log
```

Answer Key:

```
Help!
```

(b) Write a MIPS program that prints: `Hi!  Hi!`

Answer Key:

```
# Store 'Hi mom' at the top of the stack
ADDI $sp, $sp, -7
ADDI $t0, $zero, 72 # H
SB $t0, 0($sp)
ADDI $t0, $zero, 105 # i
SB $t0, 1($sp)
ADDI $t0, $zero, 33 # !
SB $t0, 2($sp)
ADDI $t0, $zero, 32 # (space)
SB $t0, 3($sp)
ADDI $t0, $zero, 72 # H
SB $t0, 4($sp)
ADDI $t0, $zero, 105 # i
SB $t0, 11($sp)
ADDI $t0, $zero, 33 # !
SB $t0, 2($sp)
ADDI $t0, $zero, 0 # (null)
SB $t0, 12($sp)

ADDI $v0, $zero, 4 # 4 is for print string
```

```
ADDI $a0, $sp, 0
syscall  # print to the log
```

9. What is the output of the following C++ programs?

(a)
```
//Lewis Carroll, Alice in Wonderland
#include <iostream>
using namespace std;
int main()
{
   cout << "Take care of the sense, "<< endl;
   cout << "and the sounds will \n take care";
   cout << "of themselves." << endl
   cout <<  endl;
}
```

Answer Key:

```
Take care of the sense,
and the sounds will
take care of themselves.
```

(b)
```
//Lewis Carroll, more Alice...
#include <iostream>
using namespace std;
int main()
{
   int count = 2;
   while (count > 0) {
      cout <<"Twinkle, ";
      count--;
   }
   cout << "little bat!"
}
```

Answer Key:

```
Twinke, Twinke, little bat!
```

```
//Stars and more stars
#include <iostream>
using namespace std;
int main()
{
  int i, j;
  for (i = 1; i <= 5; i++)
  {
    for (j = 1; j <= i; j++)
      cout << "*";
    cout << endl;
  }
}
```

(c)

Answer Key:

```
*
**
***
****
*****
```

10. (a) Write a **complete Python program** that prompts the user for a string and then prints the string in reverse. For example, if the user entered, `Python`, your program would print: `nohtyP`.

Answer Key: Many ways to do this program. Here's one:

```
#Reversing program:

s = input('Enter a string')
newS = ""
for c in s:
    newS = c + newS
print(newS)
```

Here's another

```
#Another Reversing program:

s = input('Enter a string')
newS = ""
for i in range(len(s)-1, -1,-1):
    newS = newS + s[i]
print(newS)
```

(b) Write a **complete C++ program** that prints the spread of disease, following the Susceptible, Infected, Recovered (SIR) model:

$$
\begin{aligned}
S &= .95S \\
I &= I + .05S - .04I \\
R &= R + .04I
\end{aligned}
$$

where S is the size of the susceptible population, I is the number of currently infected in the population, and R is the number who have recovered. Each day, 4% of those ill recover and 5% of the susceptible population becomes infected. Assume that the starting susceptible population at year 0 is 1000 with 100 infected and 100 recovered. Your program should print for the first 10 days: the day, the number of susceptible population, the number currently infected, and the number who have recovered.

Answer Key:

```cpp
//Checks input for positive number
#include <iostream>
using namespace std;
int main()
{
  float S = 1000, I = 100, R = 100;
  int day;
  cout << "Day\tSusceptible\tInfected\tRecovered\n";
  for (day = 0; day < 10; day++) {
    cout << day << "\t" << S << "\t" << I << "\t" << R << "\n";
    S = .95*S;
    I = I + 0.05*S - 0.04*I;
    R = R + 0.04I;
  }
  return 0;
}
```