Final Exam, Version 1
CSci 127: Introduction to Computer Science
Hunter College, City University of New York

13 December 2017

Exam Rules

- Show all your work. Your grade will be based on the work shown.

- The exam is closed book and closed notes.

- When taking the exam, you may have with you pens, pencils, and an 8 1/2" x 11" piece of paper filled with notes, programs, etc.

- You may not use a computer, calculator, tablet, smart watch, or other electronic device.

- Do not open this exam until instructed to do so.

I understand that all cases of academic dishonesty will be reported to the Dean of Students and will result in sanctions.
Name:
EmpID:
Signature:

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

```python
s = "FridaysSaturdaysSundays"
num = s.count("s")
days = s[:-1].split("s")
print("There are", num, "fun days in a week")
mess = days[0]
print("Two of them are", mess, days[-1])
result = ""
for i in range(len(mess)):
    if i > 2:
        result = result + mess[i]
print("My favorite", result, "is Saturday.")
```

Output:

(b) Consider the following shell command and resulting output:

```
ls *.html
closestCUNY.html     nycMap.html        t.html          th.html
cunySenior.html      recyc.html         tc.html         trash.html
```

   i. What is the output for:

Output:

```
ls *p.html
```

   ii. What is the output for:

Output:

```
ls *.html | grep r | grep e
```

2. (a) After executing the Python code, write the name of the turtle:

```
import turtle
turtle.colormode(255)

dennis = turtle.Turtle()
dennis.color(0,255,0)

matt = turtle.Turtle()
matt.color(0,0,0)

tim = turtle.Turtle()
tim.color("#CCCCCC")

harmon = turtle.Turtle()
harmon.color("#110000")
```

i. which is black:

ii. which is pink:

iii. which is the brightest green:

iv. which is gray:

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

```
function  makeLowerCase(inMsg)
    create an empty message
    for each letter in inMsg:
        code = the Unicode of the letter
        if code <= 90
            code = code + 32
        convert the code to the corresponding Unicode character
        concatenate the character to the beginning of the message
    return the message
```

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

    i.
```
in1 = False
in2 = True
out = in1 or in2
```
out = [ ]

    ii.
```
in1 = False
in2 = False
out = not in1 and (in1 or in2)
```
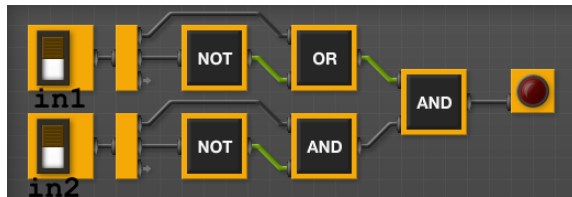out = [ ]

    iii.
```
in1 = True
in2 = False
in3 = (in1 and in2)
out = in1 or not in3
```
out = [ ]

    iv.


```
in1 = True
in2 = True
```
out = [ ]

(b) Design a circuit that takes three inputs that:
- returns false if all three inputs are false, and
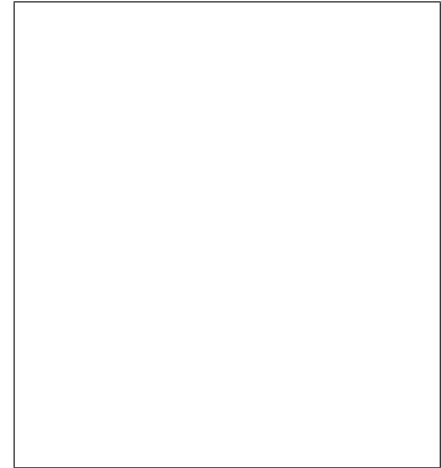- returns true otherwise.

4. (a) Draw the output of the program:

```
#Mystery program...
import turtle

th = turtle.Turtle()

for i in range(5):
    th.forward(100)
    th.left(360/5)
```

Output:

(b) What is the output:

```
#Mystery program

def select(nums):
    m = nums[0]
    for n in nums:
        if n > m:
            m = n
            print(m)
    return(m)

def truncate(userList):
    if len(userList) < 4:
        best = select(userList)
    else:
        best = select(userList[:4])
    print("Best is", best)
```

i. For `truncate([10,20])`?

Output:

ii. For `truncate([1,3,5,4])`?

Output:

iii. For `truncate([1,2,3,4,100])`?

Output:

5. Write a **complete Python program** that

- asks the user for the name of a png file and
- prints the number of pixels that are bright red (the fraction of red is above 0.75 and the fraction of green, and the fraction of blue are below 0.25).

6. Using `folium`, write a **complete Python program** that asks the user for name of the output file, and creates a map with markers for the following locations:

   - Hunter College (latitude: 40.768731 and longitude: -73.964915)
   - Empire State Building (latitude: 40.748441 and longitude: -73.985664)
   - Statue of Liberty (latitude: 40.689249 and longitude: -74.0445)

   Each marker should include a pop-up message with the name of the location.

7. Fill in the following functions that creates a graph of the fraction of population over time:

- `getData()`: asks the user for the name of the CSV and returns a DataFrame of the contents,
- `makeFraction()`: creates a column of the fraction of the borough population, and
- `makeGraph()`: makes a graph of the x versus y columns specified.

```
import pandas as pd
import matplotlib.pyplot as plt

def getData():
    """
    Asks the user for the name of the CSV. Returns a DataFrame of the contents.
    """
```

```
def makeFraction(df,top,total,frac):
    """
    Makes a new column, frac, of df that is df[top]/df[total]
    Returns the DataFrame, df
    """
```

```
def makeGraph(df,xCol,yCol):
    """
    Makes a pyplot plot of x versus y column in DataFrame df
    """
```

8. (a) What are the values of register, `$s0` for the run of this MIPS program:

```
#Sample program that loops from 20 down to 0
ADDI $s0, $zero, 20 #set s0 to 20
ADDI $s1, $zero, 5  #use to decrement counter, $s0
AGAIN: SUB $s0, $s0, $s1
BEQ $s0, $zero, DONE
J AGAIN
DONE:  #To break out of the loop
```

Values of $s0:

(b) Write a MIPS program where the register, `$s0` loops through the values: `2,4,6,8,10`

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

(a)
```cpp
//William Bulter Yeats
#include <iostream>
using namespace std;
int main()
{
    cout << "Education is not " << endl;
    cout << "the filling of a pail,\n but ";
    cout << "the lighting of a fire.\n";
}
```

Output:

(b)
```cpp
//Mystery C++, #2
#include <iostream>
using namespace std;
int main()
{
    float count = 8.0;
    while (count >  2) {
        cout << count << "\n";
        count = count/2;
    }
    count << "Boom!\n";
}
```

Output:

(c)
```cpp
//Mystery C++, #3
#include <iostream>
using namespace std;
int main()
{
    for (int i = 0; i < 5; i++) {
        for (int j = 5; j > i; j--) {
            if (j % 2 == 1)
                cout << "+";
            else
                cout << "-";
        }
        cout << endl;
    }
}
```

Output:

10. (a) Write a **complete Python program** to print the fine for speeding. The program must read the speed from user input, then compute and print the fine. The fine is $15 for each mph over 60 and less than or equal to 70, and $20 for each additional mph over 70.

    For example, if the speed is 63 mph, then the fine would be $45 = $15 x 3. If the speed is 72 mph, then the fine would be $190 = $15 x 10 + $20 x 2.

    (b) Write a **complete C++ program** that repeatedly prompts the user for the year they were born until they enter a number that is 2017 or smaller. Your program should print out the final number the user entered: