Final Exam, Version 2
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:

```
s = "history.biology.science."
num = s.count(".")
subjects = s[:-1].split(".")
print("There are", num, \
        "important subjects in school.")
mess = subjects[:-1]:
for item in mess:
    print("Don't know much about", item)
print("But I do know that", \
        "I love computer " + subjects[2])
```

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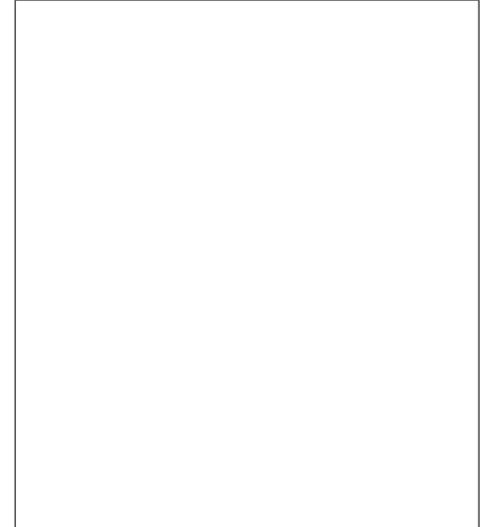
(b) Consider the following shell command and resulting output:

```
ls *.py
averageImage.py          copenhagenTransit.py      gcContent.py
averageImageAnswer.py    cropImage.py              growingSpiral.py
blueImage.py             cropImage2.py             triangles.py
```

   i. What is the output for:

Output:

```
ls *wing*.py
```

   ii. What is the output for:

Output:

```
ls *.py | grep age
```

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

```
import turtle
turtle.colormode(255)

amy = turtle.Turtle()
amy.color("#FFFFFF")

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

meg = turtle.Turtle()
meg.color("#FF0000")

lisa = turtle.Turtle()
lisa.color(100,100,100)
```

i. which is white:

ii. which is gray:

iii. which is teal (blue-green):

iv. which is brightest red:

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

```
function  makeUpperCase(inMsg)
    create an empty message
    for each letter in inMsg:
        code = the Unicode of the letter
        if code >= 97
            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 = True
in2 = False
out = in1 or in2
```
out = ⬚

ii.
```
in1 = True
in2 = False
out = in1 and not (in1 or in2)
```
out = ⬚

iii.
```
in1 = True
in2 = True
in3 = (in1 or in2)
out = in1 and not in3
```
out = ⬚

iv.
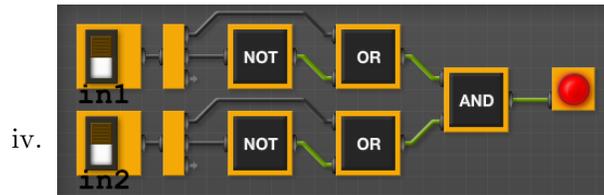
```
in1 = False
in2 = True
```
out = ⬚

(b) Design a circuit that takes three inputs that:

- returns true if all three inputs are true, and
- returns false 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(720/5)
```

Output:
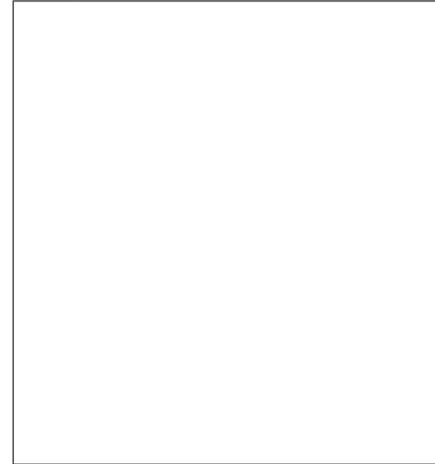
(b) What is the output:

```
#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) < 5:
        best = select(userList)
    else:
        best = select(userList[2:])
    print("Best is", best)
```

i. For `truncate([10,2])`?

Output:

ii. For `truncate([7,2,0,1])`?

Output:

iii. For `truncate([0,2,10,9,1,-1])`?

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 very purple (the fraction of red and the fraction of blue are both above 0.75 and the fraction of green is below 0.25).

6. Write a **complete Python program** that will read:

- prompt the user for the name of a CSV file,
- prompt the user for the name of a column in that CSV file, and
- print out the minimum and maximum values of that column.

7. Fill in the following functions that maps GIS data from NYC OpenData CSV files:

- getLocale(): asks and returns the user for latitude and longitude,
- makeMap(): returns a map, centered at the current location, and
- addMarkers(): adds markers, from the GIS locations in a DataFrame to the map.

```
import pandas as pd
import folium
def getLocale():
    """
    Asks the user for latitude and longitude of the user's current location and
    Returns those floating points numbers.
    """
```

```
def makeMap(x,y):
    """
    Makes a map, m, centered at (x,y) and
    Returns m
    """
```

```
def addMarkers(m,df):
    """
    Adds to map, m, markers for locations in columns df['Lat'] and df['Lon']
    Returns m
    """
```

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

```
#Sample program that loops from 50 down to 0
ADDI $s0, $zero, 50 #set s0 to 50
ADDI $s1, $zero, 10  #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:
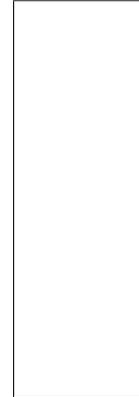
(b) Write a MIPS program where the register, $s0 loops through the values: `1,3,5,7`

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

(a)
```cpp
//Derek Bok
#include <iostream>
using namespace std;
int main()
{
    cout << "If you think education" << endl;
    cout << " is expensive,\n Try ";
    cout << "ignorance.\n";
}
```

Output:

(b)
```cpp
//Mystery C++, #2
#include <iostream>
using namespace std;
int main()
{
  int count = 0;
  while (count < 4) {
    cout << count<< endl;
    count = count + 2;
  }
  cout<<"Up!\n";
}
```

Output:

(c)
```cpp
//Mystery C++, #3
#include <iostream>
using namespace std;
int main()
{
  for (int i = 0; i < 5; i++) {
    for (int j = 0; j < 5; j++)
      if ((i+j) % 2 == 0)
        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 \$10 for each mph over 55 and less than or equal to 65, and \$15 for each additional mph over 65.

For example, if the speed is 58 mph, then the fine would be \$30 = \$10 x 3. If the speed is 67 mph, then the fine would be \$130 = \$10 x 10 + \$15 x 2.

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