

FINAL EXAM, VERSION 1  
CSci 127: Introduction to Computer Science  
Hunter College, City University of New York

13 December 2017

**Answer Key:**

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

```
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.")
```

**Answer Key:**

There are 3 fun days in a week  
Two of them are Friday Sunday  
My favorite day is Saturday.

- (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:  
ls \*p.html

**Answer Key:**

```
nycMap.html
```

- ii. What is the output for:  
`ls *.html | grep r | grep e`

**Answer Key:**

```
cunySenior.html  
recyc.html
```

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

**Answer Key:**

```
matt
```

```
import turtle  
turtle.colormode(255)
```

- ii. which is pink:

**Answer Key:**

```
harmon
```

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

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

- iii. which is the brightest green:

**Answer Key:**

```
dennis
```

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

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

- iv. which is gray:

**Answer Key:**

```
cherry
```

- (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
```

**Answer Key:**

```
def makeLowerCase(inMsg):
    mess = ""
    for letter in inMsg:
        code = ord(letter)
        if code <= 90:
            code = code + 32
        ch = chr(code)
        mess = mess + ch
    return(mess)
```

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

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

**Answer Key:**

```
out = True
```

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

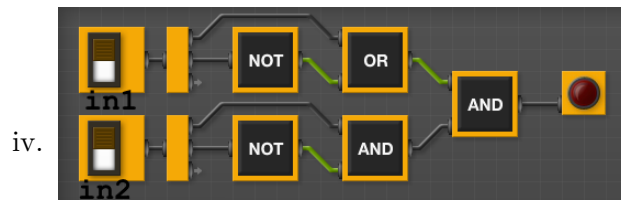
**Answer Key:**

```
out = True
```

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

**Answer Key:**

```
out = True
```



in1 = True

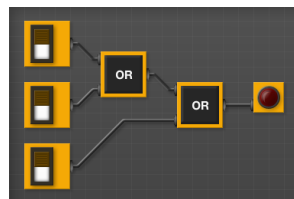
in2 = True

**Answer Key:**

out = False

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

- returns false if all three inputs are false, and
- returns true otherwise.



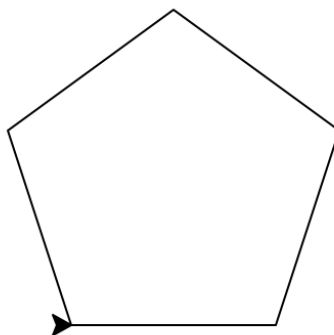
**Answer Key:**

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)
```



**Answer Key:**

(b) What is the output:

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

**Answer Key:**

20

Best is 20

#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)
```

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

**Answer Key:**

3

5

Best is 5

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

**Answer Key:**

2

3

4

Best is 4

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).

**Answer Key:**

```
#Name:  CSci 127 Teaching Staff
```

```
#Date:  Fall 2017
```

```
#Count number of red pixels in an image
```

```
#Import the packages for images and arrays:
```

```
import matplotlib.pyplot as plt
```

```
import numpy as np
```

```

fileName = input('Enter file name: ')
img = plt.imread(fileName)    #Read in image
countRed = 0                  #Number of pixels that are bright red

#For every pixel:
for i in range(img.shape[0]):
    for j in range(img.shape[1]):
        if (img[i,j,0] > 0.75) and (img[i,j,1] < 0.25) and (img[i,j,2] < 0.25):
            countRed = countRed + 1

print("Red count is", countRed)

```

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.

#### Answer Key:

```

#Map locations:
import folium

outF = input('Enter output file: ')

mapNYC = folium.Map(location=[40.768731, -73.964915], tiles="Cartodb Positron", zoom_start=12)

hc = folium.Marker([40.768731, -73.964915], popup="Hunter College")
hc.add_to(mapNYC)
esb = folium.Marker([40.748441, -73.985664], popup="Empire State Building")
esb.add_to(mapNYC)
sol = folium.Marker([40.689249, -74.0445], popup="Statue of Liberty")
sol.add_to(mapNYC)

mapNYC.save(outfile=outF)

```

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.

#### Answer Key:

```
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.
    """
    fName = input('Enter current latitude: ')
    df = pd.read_csv(fName)
    return(df)

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

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

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:**

**Answer Key:**

20  
15  
10  
5  
0

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

**Answer Key:**

```
#Program that loops from 2 up to 10, by twos
ADDI $s0, $zero, 2 #set s0 to 2
ADDI $s1, $zero, 2 #use to increment counter, s0
ADDI $s2, $zero, 10 #set s2 to use for comparison
AGAIN: ADD $s0, $s0, $s1
BEQ $s0, $s2, DONE
J AGAIN
DONE: #To break out of the loop
```

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

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

**Answer Key:**

```
Education is not
the filling of a pail,
but the lighting of a fire.
```

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

**Answer Key:**

```
8.0
4.0
Boom!
```



```

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

```

**Answer Key:**

```

+--+
+--+
+--+
+-
+

```

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.

**Answer Key:**

```

def answer1():
    speed = eval(input("Enter the speed in mph:"))
    if (speed < 60):
        print("No fine")
    else:
        fine = (speed - 60) * 15
        if speed > 70:
            fine = fine + (speed - 70) * 5
        print("The fine is", fine)

```

- (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:

**Answer Key:**

```

//Checks input for positive number

```

```
#include <iostream>
using namespace std;
int main()
{
    int year;
    cout << "Please enter the year you were born: ";
    cin >> year;
    while (year > 2017) {
        cout << "You entered a year in the future.\n";
        cout << "Please enter the year you were born: ";
        cin >> year;
    }
    cout << "Year your were born: " << year;
    return 0;
}
```