

FINAL EXAM, VERSION 4
CSci 127: Introduction to Computer Science
Hunter College, City University of New York

20 December 2017

Answer Key:

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

```
wList = "Mon&Tues&Wednes&Thurs&Fri&Satur&Sun"
weeks = wList.split("&")
tgit = weeks[3]
print(tgit.upper())
print(weeks[-1])
for i in range(0,6,2):
    day = weeks[i]+"day"
    print(i,day)
```

Answer Key:

```
THURS
Sun
0 Monday
2 Wednesday
4 Friday
```

- (b) Consider the following shell commands:

```
$ ls
map.html  map.py  p10.py
```

- i. What is the output for:
- ```
$ cp p10.py p11.py
$ mkdir maps
$ ls
```

**Answer Key:**

map.html map.py maps/ p10.py p11.py

ii. What is the output for:

```
$ mv p10.py map2.py
$ mv m*. * maps
$ ls
```

**Answer Key:**

maps/ p11.py

2. (a) Fill in the missing values in the table:

| Decimal                | Binary                 | Hexadecimal           |
|------------------------|------------------------|-----------------------|
| 4                      | <b>Answer Key: 100</b> | 4                     |
|                        |                        |                       |
| <b>Answer Key: 5</b>   | 101                    | 5                     |
|                        |                        |                       |
| 14                     | 1110                   | <b>Answer Key: E</b>  |
|                        |                        |                       |
| 32                     | 100000                 | <b>Answer Key: 20</b> |
|                        |                        |                       |
| <b>Answer Key: 255</b> | 11111111               | FF                    |
|                        |                        |                       |

(b) Fill in the code below to make an image in which a pixel is blue if it has an entry of 10 or less in the array `elevations`. Otherwise, the pixel should be colored green.

```
Takes elevation data of NYC and displays storm surge map
import numpy as np
import matplotlib.pyplot as plt
elevations = np.loadtxt('elevationsNYC.txt')
#Base image size on shape (dimensions) of the elevations:
mapShape = elevations.shape + (3,)
floodMap = np.zeros(mapShape)
```

```
for row in range(mapShape[0]):
 for col in range(mapShape[1]):
```

**Answer Key:**

```
 if elevations[row,col] <= 10:
 #Below sea level
 floodMap[row,col,2] = 1.0 #Set the blue channel to 100%
 else:
 #Above sea level
 floodMap[row,col,1] = 1.0 #Set the green channel to 100%

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

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

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

**Answer Key:**

```
out = False
```

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

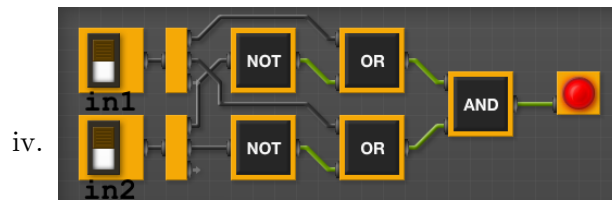
**Answer Key:**

```
out = True
```

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

**Answer Key:**

```
out = False
```



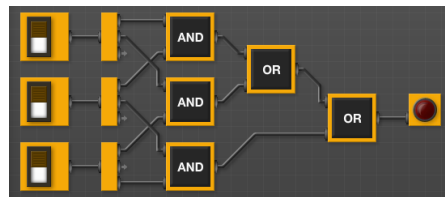
in1 = False

in2 = True

**Answer Key:**

out = True

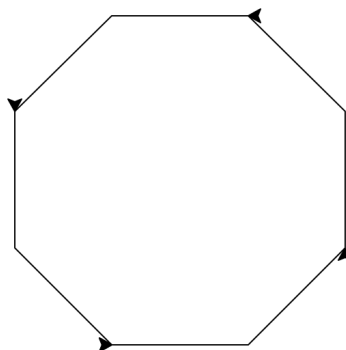
- (b) Design a circuit that takes three inputs and returns true when two or more of the inputs are true. Otherwise if returns false.



**Answer Key:**

4. (a) Draw the output of the program:
- ```
#turtle mystery
import turtle
```

```
tess = turtle.Turtle()
for i in range(8):
    if i%2 == 0:
        tess.stamp()
        tess.forward(100)
        tess.left(45)
```



Answer Key:

- (b) What is the output:

```
#Another mystery program...
```

```
#mystery
```

```
def dub(num):
```

```
    send = chr(num)
```

```
    if num < ord('k'):
```

```
        send = send + send
```

```
    return send
```

```
def enigma(letters):
```

```
    mess = ""
```

```
    for l in letters:
```

```
        n = ord(l)
```

```
        c = dub(n)
```

```
        mess = mess + c
```

```
    return mess
```

```
word = input('Enter a word: ')
s = enigma(word)
print('Output is:', s)
```

i. When the user enters: aa?

Answer Key: Output is: aa

ii. When the user enters: cab?

Answer Key: Output is: ccaabb

iii. When the user enters: alice?

Answer Key: Output is: aaliiccee

5. 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 maximum value and the average of the column.

Answer Key:

```
#Computes average and standard deviation of inputted column
import pandas as pd
```

```
fileName = input('Enter file name: ')
colName = input('Enter column name: ')
df = pd.read_csv(fileName)
M = df[colName].max()
ave = df[colName].mean()
print("Maximum is ", M)
print("Average is ", ave)
```

6. Write a **complete Python program** that asks the user for the name of a .png (image) file and displays the lower left quarter of the image.

For example if the image is `hunterLogo.png` (left), the displayed image would be (right):



Answer Key:

```
#Name:  CSci 127 Teaching Staff
#Date:  Fall 2017
#This program loads an image, displays it, and then creates and displays
#      a new image that is only the upper left corner.

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

inF = input('Enter file name: ')
img = plt.imread(inF)    #Read in image from inF

height = img.shape[0]    #Get height
width = img.shape[1]     #Get width
print(height,width)

img2 = img[height/2:, :width/2]    #Crop to upper right corner

plt.imshow(img2)           #Load our new image into pyplot
plt.show()                 #Show the image (waits until closed to continue)
```

7. Complete the following Python program, which asks the user for the number of turtles to create and creates them. Then, repeats 10 times, moving each turtle in the list forward and stamping. That is, write the functions `getInput()`, `moveForward()`, and `stamp()`:

```
#Intro Programming Lab:  A program with herd of turtles
import turtle
def setUpTurtles(n):
    tList = []
    #Create turtles:
    for i in range(n):
        t = turtle.Turtle()
        t.shape("turtle")
        tList.append(t)
    return tList
def main():
    num = getInput()          #Ask for number of turtles
    turtleList = setUpTurtles(numTurtles) #Make a list of num turtles
    for i in range(10):      #Repeat 10 times
        moveForward(turtleList) #Move each turtle in the list forward
        stamp(turtleList)      #Stamp where the turtle stopped
if __name__ == "__main__":
    main()
```

Answer Key:

```

def getInput():
    n = eval(input("Please enter the number of turtles: "))
    return n

def moveForward(tList):
    for t in tList:
        t.forward(30)

def stamp(tList):
    for t in tList:
        t.stamp()

```

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

```

#Set up counters:
ADDI $s0, $zero, 3
ADDI $s1, $zero, 1
AGAIN: ADDI $sp, $sp, -4
ADDI $t0, $zero, 76 # B
SB $t0, 0($sp)
ADDI $t0, $zero, 121 # y
SB $t0, 1($sp)
ADDI $t0, $zero, 101 # e
SB $t0, 2($sp)
ADDI $t0, $zero, 0 # (null)
SB $t0, 3($sp)
ADDI $v0, $zero, 50 #50 is for printing message
ADDI $a0, $sp, 0
syscall
SUB $s0, $s0, $s1
BEQ $s0, $zero, AGAIN

```

Answer Key: Values of \$s0:

2
1
0

Answer Key: Output:

Bye
Bye

- (b) Write a MIPS program that prints the letter 'B' 10 times:

Answer Key:

```

#Set up counters:
ADDI $s0, $zero, 10
ADDI $s1, $zero, 1
AGAIN: ADDI $sp, $sp, -2
ADDI $t0, $zero, 76 # B
SB $t0, 0($sp)
ADDI $t0, $zero, 0 # (null)
SB $t0, 1($sp)
ADDI $v0, $zero, 50 #50 is for printing message
ADDI $a0, $sp, 0
syscall
SUB $s0, $s0, $s1
BEQ $s0, $zero, AGAIN

```

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

```

//Eleanor Roosevelt
#include <iostream>
using namespace std;
int main()
(a) {
    cout << "Do one thing, every" << endl;
    cout << "day,\n that";
    cout << "scares you.\n";
}

```

Answer Key:

Do one thing, every day,
that scares you.

```

//Mystery C++, #2
#include <iostream>
using namespace std;
int main()
{
(b)   int sum = 1;
      while (sum < 10) {
          cout << sum;
          sum = sum + sum;
      }
}

```

Answer Key:

1
2


```

4
8

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

```

Answer Key:

```

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

```

10. (a) Write a **complete Python program** that prompts the user to the year they were born. If the user enters a number larger than 2017, your program should continue prompting the user until they enter a number less than or equal to 2017. Your program should then print out the number entered.

Answer Key:

```

#Asks for year, with input checking:
year = 2018
while year > 2017:
    year = int(input('Enter the year you were born ( <= 2017): '))

print("You entered", year)

```

- (b) Write a **complete C++ program** that asks the user for a number and prints “Days” if the number entered is less than 7, “Week” if the number is exactly 7, and “More” otherwise.

Answer Key:

```

//Checks input for positive number
#include <iostream>
using namespace std;

```

```
int main()
{
    int number;

    cout << "Please enter number: ";
    cin >> number;
    if (number < 0)
        cout << "Days\n";
    else if (number == 0)
        cout << "Week\n";
    else
        cout << "More\n";
    return 0;
}
```