

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

14 May 2019

Answer Key:

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

```
s = "Cersei,Lannister;Daenerys,Targaryen;Margaery,Tyrell;Yara,Greyjoy"
i. queens = s.split(';')
   t = queens[1].split(',')
   print(t.upper())
```

Answer Key:

DAENERYS

```
b,c = queens[1],queens[3]
ii. print(c[-3:])
    print(b[-6:-2])
```

Answer Key:

joy
gary

```
for q in queens:
iii.   w = q.split(',')
       print(w[1],w[0][0],'.')
```

Answer Key:

Lannister C .
Targaryen D .
Tyrell M .
Greyjoy Y .

- (b) Consider the following shell commands:

```
$ ls -l
drwxr-xr-x  32 stjohn  staff      1088 May 8  2019 drafts/
-rwxrwxrwx@  1 stjohn  staff    1136855 May 4  2019 finalS19V1.pdf*
-rwxrwxrwx@  1 stjohn  staff    1125569 May 4  2019 finalS19V2.pdf*
-rw-r--r--@  1 stjohn  staff     246352 May 5  2019 mapFinal.pdf
-rw-r--r--@  1 stjohn  staff     571936 May 2  2019 mapFinalCropped.jpg
drwxr-xr-x  21 stjohn  staff        714 May 3  2019 sign-in/
drwxr-xr-x   7 stjohn  staff         238 May 8  2019 submittedS19/
```

- i. What is the output for:

```
$ ls *S19*
```

Answer Key:

```
finalS19V1.pdf
finalS19V2.pdf
submittedS19
```

- ii. What is the output for:

```
$ ls *S19* | wc -w
```

Answer Key:

3

- iii. What is the output for:

```
$ ls -l | grep "May" | wc -w
```

Answer Key:

7

2. (a) Fill in the boxes with the appropriate hexcode to change the color to match the comments:

Answer Key:

```
import turtle
```

```
thomasH = turtle.Turtle()
```

- i. #Change thomasH to be the color black:

```
thomasH.color("# 0 0 0 0 0 0 ")
```

- ii. #Change thomasH to be the color white:

```
thomasH.color("# F F F F F F ")
```

- iii. #Change thomasH to be the brightest color blue:

```
thomasH.color("# 0 0 0 0 F F ")
```

- iv. #Change thomasH to be the color purple:

```
thomasH.color("# F F 0 0 F F ")
```

v. #Change thomasH to be the color gray:

```
thomasH.color("#191919")
```

(All grays have the same value for red, green, and blue. So, you can use any 2-digit hex number in place of "19.")

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

Ask user for input, and store in the string, hexString.

Set decNum = 0.

For each c in hexString,

Set n to be ord(c)

If n is between 48 and 57, set n to be n - ord('0').

Otherwise, set n to be n - ord('A') + 10.

Multiply decNum by 16 and add n to it (decNum = 16 * decNum + n).

Print decNum.

Answer Key:

```
#hexadecimal to decimal
```

```
hexString = input('Enter num: ')
decNum = 0
```

```
for c in hexString:
```

```
    n = ord(c)
```

```
    if 48 <= n <= 57:    #It's the ASCII for '0' to '9'
```

```
        n = n - ord('0')
```

```
    else:                #It's the ASCII for 'A' to 'F'
```

```
        n = n - ord('A') + 10
```

```
    decNum = decNum * 16 + 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 and (in2 or not in1)
```

Answer Key:

```
out = True
```

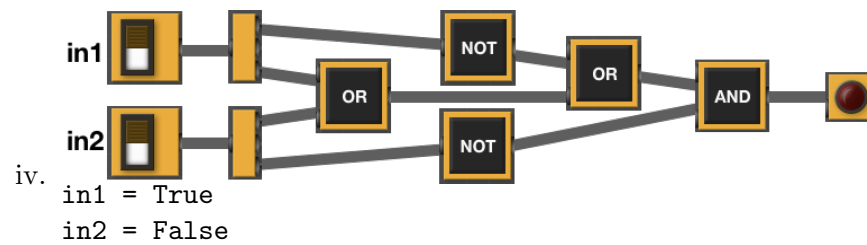
```

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

```

Answer Key:

out = True



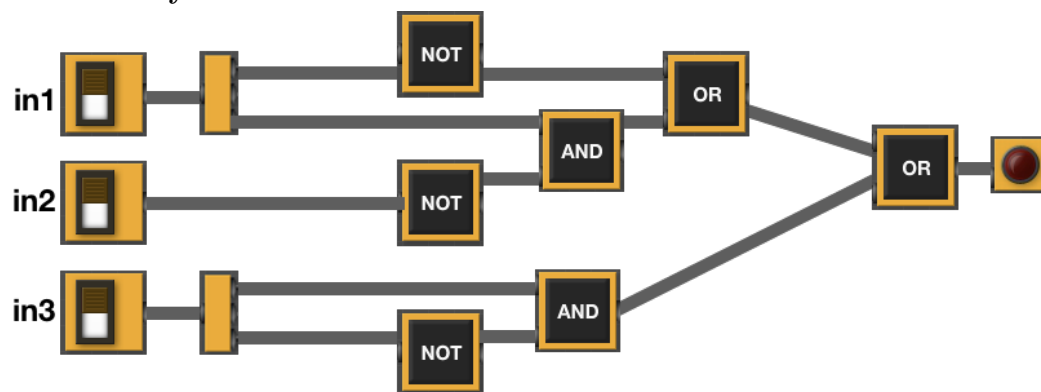
Answer Key:

out = True

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

$$((\text{not } in1) \text{ or } (in1 \text{ and not } in2)) \text{ or } (in3 \text{ and not } in3)$$

Answer Key:



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

i. `ramble(tess,50,False)`

Answer Key:

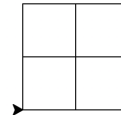
```
import turtle
tess = turtle.Turtle()

def ramble(t, len, isNested):
    if len >= 50:
        for i in range(4):
            t.forward(len)
            t.left(90)
            if isNested:
                ramble(t, len-50, isNested)
```



ii. `ramble(tess,100,True)`

Answer Key:



(b) For the following code:

```
def myst(tommi, rhia):
    if rhia < 5:
        return rhia
    else:
        return tommi
```

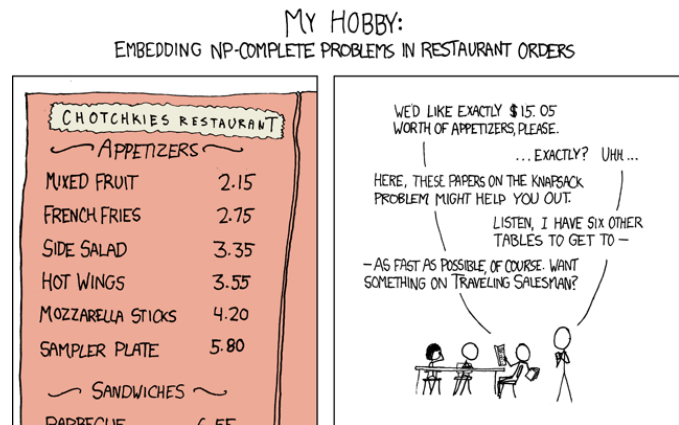
```
def start(shantel):
    mandy = 8
    savannah = myst(mandy, shantel)
    return savannah
```

i. What are the formal parameters for `myst()`: **Answer Key:** `tommi, rhia`

ii. What are the formal parameters for `start()`: **Answer Key:** `shantel`

iii. What does value does `start(10)` return: **Answer Key:** `8`

5. (a) For the menu to the right, if there is an appetizers order that will total to exactly the amount \$15.05, write it below. If there isn't, write "NO ORDER."



xxcd, #278; Alt-Text: General solutions get you a 50% tip.

- (b) Design an algorithm to solve this, for any restaurant and any dollar amount:

- **Input:**

Answer Key: A menu (with prices) and a dollar amount.

- **Output:**

Answer Key: If an order exists, the items in the order, or "NO ORDER" if none exists.

- **Process:**

Answer Key: There's several different ways to approach this. All are variations on the theme, of "try anything." Here's one approach:

- For each item on the menu, if its price divides evenly into the amount, print out how many times and that item.
- If no item evenly divides the amount, try all combinations of pairs of items (1 of first, 1 of second; 1 of first, 2 of second; 2 of first, 1 of second;...). If there's a solution, print it out.
- If none of those work, try with all triples of items. If there's a solution, print it out.
- Continue until you have tried combination of all items. If there's a solution, print it out.
- If not solution, print out "NO ORDER"

6. **Fill in the comments** to describe what each line of code does:

Answer Key:

```
#Includes the package used for creating HTML maps
import folium
```

```

#Include the library for generating random numbers
import random
#Include the pandas library for data analysis
import pandas as pd

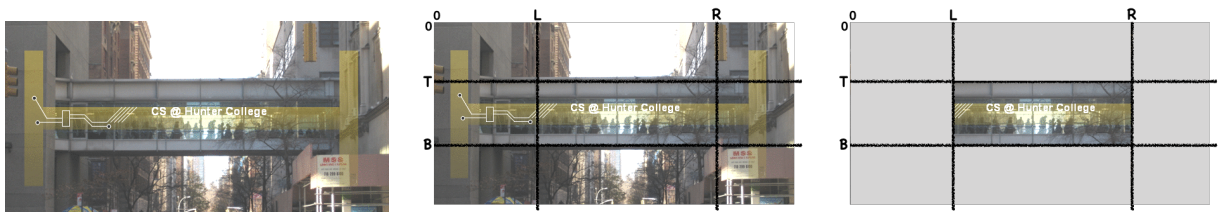
#Read in the landmarks CSV file into a dataframe, called landmarks:
landmarks = pd.read_csv('nycLandmarks.csv')
#Choose a random whole number between 0 and 3:
r = random.randint(0,4)
#Find the name for that entry in our data file:
randName = landmarks['Name'][r]
#Find the latitude for that entry in our data file:
randLat = landmarks['Latitude'][r]
#Find the longitude for that entry in our data file:
randLon = landmarks['Longitude'][r]

#Set up a new map object, centered at Hunter College, zoomed in enough to see city outline
map = folium.Map(location=[40.75, -74.125], zoom_start=10)
#Add a marker for our randomly chosen landmark to the map.
mark = folium.Marker(location = [randLat, randLon], popup = randName)
mark.add_to(map)

#Save our map to the file, nycMap.html
map.save(outfile='nycMap.html')

```

7. Write a **complete Python program** that prompts the user for the name of an .png (image) file and the upper left and lower right coordinates (“bounding box”) and displays the image cropped to the bounding box:



Answer Key:

```

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

inImg = input('Enter input image: ')
img = plt.imread(inImg) #Read in image from csBridge.png
plt.imshow(img)         #Load image into pyplot

```

```

plt.show()                #Show the image (waits until closed to continue)

outImg = input('Enter out image: ')
t = int(input('Enter top:'))
b = int(input('Enter bottom:'))
l = int(input('Enter left: '))
r = int(input('Enter right: '))

img2 = img[t:b,l:r]      #Slice the original array by dimensions entered

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

plt.imsave(outImg, img2) #Save the image we created to the out file.

```

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

```

#Sample program that loops up to 100
ADDI $s0, $zero, -25 #set s0 to -25
ADDI $s1, $zero, 25 #use to increment counter, $s0
ADDI $s2, $zero, 100 #use to compare for branching
AGAIN: ADD $s0, $s0, $s1
BEQ $s0, $s2, DONE
J AGAIN
DONE: #To break out of the loop

```

Answer Key:

```

-25
0
25
50
75
100

```

- (b) Indicate what modifications are needed to the MIPS program (repeated below) so that it increments by 5 stopping at 0 (shade in the box for each line that needs to be changed and rewrite the instruction in the space below).

Answer Key:

```

#Sample program that loops up to 100
ADDI $s0, $zero, -25 #set s0 to -25
ADDI $s1, $zero, 25 #use to increment counter, $s0
ADDI $s2, $zero, 100 #use to compare for branching
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?

```
//M Mancina & L-M Miranda
#include <iostream>
using namespace std;
int main()
{
(a)  cout << "If the wind in my sail ";
      cout << "on the sea stays behind me";
      cout << endl << "One day I'll know, ";
      cout << "how far I'll go\n";
}
```

Answer Key:

If the wind in my sail on the sea stays behind me
One day I'll know, how far I'll go

```
//L-M Miranda-- more Moana
#include <iostream>
using namespace std;
int main()
{
  int count;
(b)  cout << "For the tides, the sun,";
      cout << "the sky\nHey, ";
      for (count = 0; count < 2; count++) {
        cout << "it's okay, ";
      }
      cout << endl << "You're welcome";
}
```

Answer Key:

For the tides, the sun,the sky
Hey, it's okay, it's okay,
You're welcome

```

//Counting
#include <iostream>
using namespace std;
int main()
{
    int i, j;
(c)  for (i = 0; i < 5; i++)
    {
        for (j = 0; j < 5; j++)
            cout << i+j << " ";
        cout << endl;
    }
}

```

Answer Key:

```

0 1 2 3 4
1 2 3 4 5
2 3 4 5 6
3 4 5 6 7
4 5 6 7 8

```

10. (a) Write a complete **Python program** that uses the turtle graphics library, creates a turtle, prompts the user for a string, and then controls the turtles actions:
- 'F': moves the turtle forward
 - 'L': turns the turtle 90 degrees to the left
 - 'R': turns the turtle 90 degrees to the right

Answer Key:

```

import turtle

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

commands = input("Please enter a command string: ")

for ch in commands:
    if ch == 'F':                #move forward
        tess.forward(50)
    elif ch == 'L':              #turn left
        tess.left(90)
    elif ch == 'R':              #turn right
        tess.right(90)
    else:                        #for any
        print("Error: do not know the command:", ch)

```

- (b) Write a **complete C++ program** that asks the total number of hours until the weekend starts, and then prints out the number of complete days and hours remaining. For example,

if the user entered, 52, the program should print: 2 days and 4 hours.

Answer Key:

```
//Checks input for year born
#include <iostream>
using namespace std;
int main()
{
    int hours;
    cout << "Enter hours until weekend: ";
    cin >> hours;
    int days = hours / 24;
    int rem = hours % 24;
    cout << days << " days and " << rem << " hours\n";
    return 0;
}
```