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

6 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 exams 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
places = "Connecticut*New York*New Jersey*Pennsylvania"
num = places.count("*")
states = places.split("*")
print("There are", num+1, "states")
print(states[0], states[-1])
mess = "cahbecedseef gchaikje"
eat = ""
for i in range(len(mess)):
    if i % 2 == 0:
        print(mess[i])
        eat = eat + mess[i]
print("I love", states[1], eat)
```

Output:

(b) Consider the following shell commands:

```
$ ls
p1.py   p2.py   p3.py
```

    i. What is the output for:

Output:

```
$ cp p1.py p4.py
$ mv p2.py p5.py
$ ls p*
```

    ii. What is the output for:

Output:

```
$ mkdir programs
$ mv p*.py programs
$ ls
```

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

Decimal	Binary	Hexadecimal
2		2
	101	5
	1010	A
32	100000	
255	11111111	

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

```
function  decodeMessage(numbers, k)
    create an empty message
    for each number in the list of numbers
        code = 97 + ((number + k) modulo 26)
        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):

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

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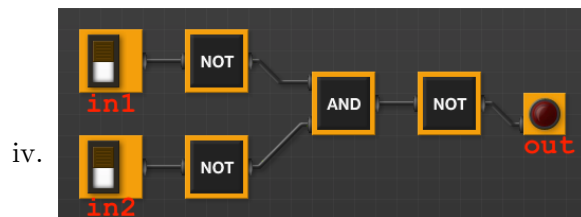out =

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

        iv.


```
in1 = True
in2 = False
```
out =

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

```
(in1 and in2) or (in1 and in3) or (in2 and in3)
```

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

```
import folium as f
mapNYC = f.Map(location=[40.768731, -73.964915],\
               tiles="cartodbpositron",\
               zoom_start=15)
#68 & Lex:
f.Marker(location = [40.768731, -73.964915],\
         popup = "Hunter College").add_to(mapNYC)
#72 & 2nd:
f.Marker(location = [40.768802, -73.958361],
         popup = "Q Train").add_to(mapNYC)
#59 & Lex:
f.Marker(location = [40.762526, -73.967967],\
         popup = "4 Train").add_to(mapNYC)
mapNYC.save(outfile='nycMap.html')
```

Output:

(b) What is returned when the function is invoked on the inputs below:

```
def enigma4(x,y,z):
    if x < y:
        return(z)
    elif x == y:
        return(z+z)
    else:
        s = cont4(z)
        return(s)
```

```
def cont4(st):
    r = ""
    for i in range(len(st)-1,-1,-1):
        r = r + st[i]
    return(r)
```

i. enigma4(3,4,"blondie")

Return:

ii. enigma4(50,50,"Yum")

Return:

iii. enigma4(10,0,"einworb")

Return:

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 containing the x values,
- prompt the user for the name of a column containing the y values, and
- make a graph (using `matplotlib.pyplot`) of the x versus y values.

6. Using `matplotlib.pyplot` and `numpy`, write a **complete Python program** that asks the user for the name of an image file, reads in the file, and displays only the "green" channel of the image to the screen.

7. Fill in the following functions that are part of a program that maps GIS data from NYC OpenData CSV files:

- `getData()`: asks the user for the name of the CSV and returns a DataFrame of the contents.
- `getLocale()`: asks the user for latitude and longitude of the user's current location and returns those floating points numbers, and
- `computeDist()`: computes the squared distance between two points (x1,y1) and (x2,y2):

$$(x1 - x2)^2 + (y1 - y2)^2$$

```
import pandas as pd
def getData():
    """
    Asks the user for the name of the CSV and
    Returns a dataframe of the contents.
    """
```

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

```
def computeDist(x1,y1,x2,y2):
    """
    Computes the squared distance between two points (x1,y1) and (x2,y2) and
    Returns (x1-x2)^2 + (y1-y2)^2
    """
```

8. (a) What is the output for a run of this MIPS program:

```
# Store 'Hello!' at the top of the stack
ADDI $sp, $sp, -7
ADDI $t0, $zero, 72 # H
SB $t0, 0($sp)
ADDI $t0, $zero, 101 # e
SB $t0, 1($sp)
ADDI $t0, $zero, 108 # l
SB $t0, 2($sp)
ADDI $t0, $zero, 108 # l
SB $t0, 3($sp)
ADDI $t0, $zero, 111 # o
SB $t0, 4($sp)
ADDI $t0, $zero, 33 # !
SB $t0, 5($sp)
ADDI $t0, $zero, 0 # (null)
SB $t0, 6($sp)
ADDI $v0, $zero, 4 # 4 is for print string
ADDI $a0, $sp, 0
syscall  # print to the log
```
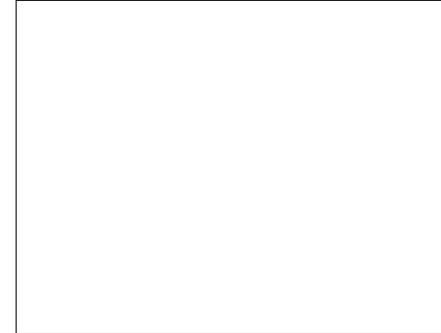
Output:

(b) Write a MIPS program that prints: Hi mom

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

(a)
```cpp
//L-M Miranda, #1
#include <iostream>
using namespace std;
int main()
{
  cout << "I wanna build\nSomething ";
  cout << "that's gonna";
  cout << "\nOutlive me" << endl;
}
```
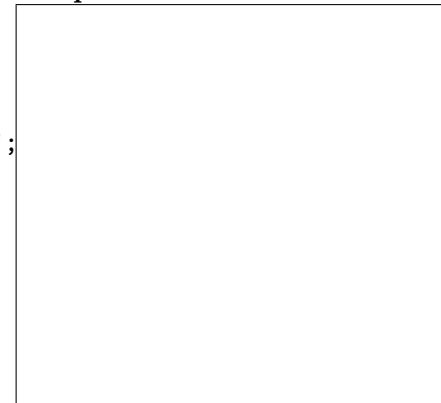
Output:

(b)
```cpp
//L-M Miranda, #2
#include <iostream>
using namespace std;
int main()
{
  int count = 4;
  while (count > 0) {
    cout <<"\tWhat do you want, Burr?\n";
    count--;
  }
}
```

Output:

(c)
```cpp
//L-M Miranda, #3
#include <iostream>
using namespace std;
int main()
{
  cout << "If we lay a strong enough found";
  cout << "ation\nWe'll pass it on to you, ";
  cout << "we'll give the world to you\n";
  for (int i = 0; i < 2; i++) {
    if (i % 2 == 0)
      cout << "And ";
    else
      cout << "Yeah, ";
    cout << "you'll blow us all away...\n";
    cout << "Someday, someday" << endl;
  }
}
```

Output:

10. (a) Write a **complete Python program** that simulates a random walk using the `turtle library`. Your turtle, at each iteration, should choose a random direction and walk forward 10 steps. The walk should continue as long as the turtle's x and y coordinate values are within 100 of the starting point.

    (b) Write a **complete C++ program** that prints out the yearly population, following the formula:
    $$p(n+1) = 1.10 \cdot p(n) - \frac{.20 \cdot (p(n))^2}{20000}$$
    where $p(n)$ is the population at year n. Assume that the starting population at year 0 is 600. Your program should print the year and population for the first 10 years.