

FINAL EXAM, VERSION 3  
CSci 127: Introduction to Computer Science  
Hunter College, City University of New York

19 December 2018

**Answer Key:**

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

```
i. s = "avram,henriette;dolciani,mary;rees,mina"  
   a = s[6:11]  
   print(a.upper())
```

**Answer Key:**

HENRI

```
ii. names = s.split(';')  
    print(names[-1])
```

**Answer Key:**

rees,mina

```
iii. b,c = names[0],names[1]  
     print(b[:5])
```

**Answer Key:**

avram

```
iv. for n in names:  
     w = n.split(',')  
     print(w[1],w[0])
```

**Answer Key:**

henriette avram  
mary dolciani  
mina rees

(b) Consider the following shell commands:

```
$ ls  
nyc.csv  p53.cpp  p54.cpp  p55.cpp  trees.csv
```

i. What is the output for:

```
$ ls *.cpp
```

**Answer Key:**

p53.cpp p54.cpp p55.cpp

ii. What is the output for:

```
$ ls *.cpp | wc -l
```

**Answer Key:**

3

iii. What is the output for:

```
$ mkdir ccProgs  
$ echo "Created folder: ccProgs"
```

**Answer Key:**

Created folder: ccProgs

2. (a) For each row below containing a binary, decimal, and hexadecimal number, circle the **largest value** in the row (or “All Equal” if all three entries have the same value):

	Binary:	Decimal:	Hexadecimal:	All Equal
a)	11	10	<b>9</b>	<i>All Equal</i>
b)	<b>111</b>	5	5	<i>All Equal</i>
c)	<b>101010</b>	32	21	<i>All Equal</i>
d)	1000000	64	40	<b>All Equal</b>
e)	11111110	254	<b>FF</b>	<i>All Equal</i>

**Answer Key:**

- (b) Fill in the code below to make an image in which a pixel is white if it has an entry of 0 in the array `elevations`. Otherwise, the pixel should be colored blue.

```
# Takes elevation data of NYC and displays coastlines
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] == 0:
            #Coastline:
            floodMap[row,col,:] = 1.0      #Set all channels to 100%
        else:
            #Everyone else
            floodMap[row,col,2] = 1.0      #Set the blue channel to 100%

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

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

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

**Answer Key:**

out = False

in1 = True

ii. in2 = True

out = not in1 or (in2 and not in1)

**Answer Key:**

out = False

in1 = False

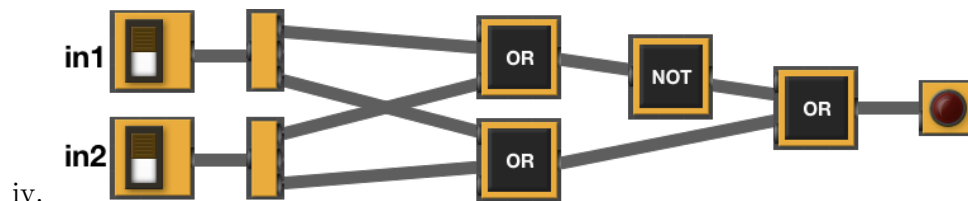
iii. in2 = True and not in1

in3 = in1 and in2

out = in1 or not in3

**Answer Key:**

out = True



in1 = True

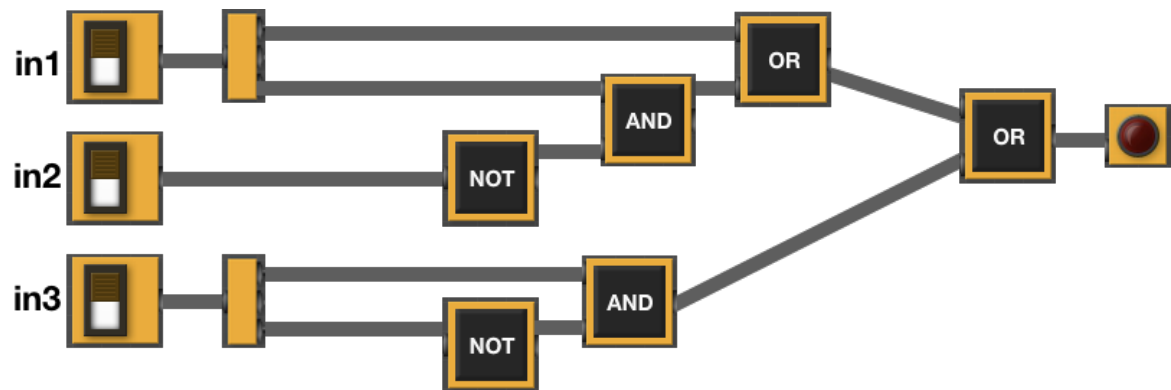
in2 = False

**Answer Key:**

out = True

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

$$(in1 \text{ or } (in1 \text{ and not } in2)) \text{ or } (in3 \text{ and not } in3)$$
**Answer Key:**



4. (a) For the following code:

```
def v3(anoop, madison):
    if anoop > madison:
        return madison
    else:
        return -1
```

```
def startV3(shelly):
    jack = 5
    rachael = 20
    alexandra = v1(jack+shelly,rachael)
    return alexandra
```

- i. What are the formal parameters for `v3()`:

**Answer Key:**

anoop, madison

- ii. What are the formal parameters for `startV3()`:

**Answer Key:**

shelly

- iii. What does `startV3(20)` return:

**Answer Key:**

20

- (b) Given the function definition:

```
def sorted(ls):
    for i in range(4):
        print(ls)
        for j in range(3):
            if ls[j] > ls[j+1]:
                ls[j],ls[j+1] = ls[j+1],ls[j]
```

- i. What is the output for `sorted([20,10,0,5])`?

**Answer Key:**

ls[0]	ls[1]	ls[2]	ls[3]
20	10	0	5
10	0	5	20
0	5	10	20
0	5	10	20

ii. What is the output for `sorted(["Nicky","Maria","Ferdi","Andrey"])?`

**Answer Key:**

ls[0]	ls[1]	ls[2]	ls[3]
"Nicky"	"Maria"	"Ferdi"	"Andrey"
"Maria"	"Ferdi"	"Andrey"	"Nicky"
"Ferdi"	"Andrey"	"Maria"	"Nicky"
"Andrey"	"Ferdi"	"Maria"	"Nicky"

5. Design an algorithm that prints out all the street trees in your zip code from the NYC Urban Forest OpenData. Specify the inputs and outputs for your algorithm and give the design in pseudocode. In your pseudocode, specify any libraries that you would need for your design.

1	created_at	tree_id	block_id	the_geom	curb_loc	status	health	spc_latin	spc_commor	address	zipcode	zip_city
2	8/27/15	180683	348711	POINT (-73.8	OnCurb	Alive	Fair	Acer rubrum	red maple	108-005 70 AVENUE	11375	Forest Hills
3	9/3/15	200540	315986	POINT (-73.8	OnCurb	Alive	Fair	Quercus pal	pin oak	147-074 7 AVENUE	11357	Whitestone
4	9/5/15	204026	218365	POINT (-73.9	OnCurb	Alive	Good	Gleditsia tria	honeylocust	390 MORGAN AVENUE	11211	Brooklyn
5	9/5/15	204337	217969	POINT (-73.9	OnCurb	Alive	Good	Gleditsia tria	honeylocust	1027 GRAND STREET	11211	Brooklyn
6	8/30/15	189565	223043	POINT (-73.9	OnCurb	Alive	Good	Tilia americana	American linden	603 6 STREET	11215	Brooklyn
7	8/30/15	190422	106099	POINT (-73.9	OnCurb	Alive	Good	Gleditsia tria	honeylocust	8 COLUMBUS AVENUE	10023	New York
8	8/30/15	190426	106099	POINT (-73.9	OnCurb	Alive	Good	Gleditsia tria	honeylocust	120 WEST 60 STREET	10023	New York
9	9/7/15	208649	103940	POINT (-73.9	OnCurb	Alive	Good	Tilia americana	American linden	311 WEST 50 STREET	10019	New York
10	9/8/15	209610	407443	POINT (-74.0	OnCurb	Alive	Good	Gleditsia tria	honeylocust	65 JEROME AVENUE	10305	Staten Island
11	8/31/15	192755	207508	POINT (-73.9	OffsetFromCurb	Alive	Fair	Platanus x acerifolia	London plane	638 AVENUE Z	11223	Brooklyn
12	9/5/15	203719	302371	POINT (-73.9	OnCurb	Alive	Good	Platanus x acerifolia	London plane	20-025 24 STREET	11105	Astoria
13	9/5/15	203726	302371	POINT (-73.9	OnCurb	Alive	Poor	Platanus x acerifolia	London plane	20-055 24 STREET	11105	Astoria
14	9/1/15	195202	415896	POINT (-74.1	OnCurb	Alive	Fair	Platanus x acerifolia	London plane	35 FENWAY CIRCLE	10308	Staten Island
15	8/30/15	189465	219493	POINT (-73.9	OnCurb	Alive	Good	Platanus x acerifolia	London plane	100 WAVERLY AVENUE	11205	Brooklyn

**Input:**

**Answer Key:** The name of the CSV file and the zip code

**Output:**

**Answer Key:** All trees (either rows or `tree_id`) of trees in that zip code.

**Process:**

**Answer Key:**

- Import pandas.
- Ask user for file name and zip code.
- Open the file as a `dataFrame`.

- (d) Select all the rows (or `tree_id`'s) where `zipcode` matches the one entered.  
(e) Print out selected rows (or `tree_id`'s).

6. Fill in the Python program that will:

- prompt the user for the name of a CSV file,
- prompt the user for the name of a column in that CSV file,
- print out the maximum value of the column, and
- displays a bar plot of the column entered (with "Year" as the x-axis).

#P6,V3: prints max of a column in a CSV file & makes a plot

#Import the libraries for data frames and displaying images:

#Prompt user for file name:

#Prompt user for column name:

`df = pd.read_csv(fileName)`

#Compute maximum value of the column:

`print("Maximum is ", M)`

#Display a bar plot of "Year" vs. column entered by user:

**Answer Key:**

```
#P6,V3: prints max of a column in a CSV file & makes scatter plot

#Import the libraries for data frames and displaying images:
import pandas as pd
import matplotlib.pyplot as plt

#Prompt user for file name:
fileName = input('Enter file name: ')

#Prompt user for column name:
col = input('Enter column name: ')

df = pd.read_csv(fileName)

#Compute maximum value of the column:
M = df[col].max()
print("Maximum is ", M)

#Display a scatter plot of "Year" vs. column entered by user:
df.plot.bar(x = "Year", y = col)
plt.show()
```

7. Complete the following program, by writing the functions:

- `getInput()`: returns the number of turtles the user entered,
- `setUp()`: sets up a graphics window and turtle, and
- `drawLines()`: repeat 10 times: n steps, turn left 60 degrees.

**Answer Key:**

```
import turtle

def getInput():
    n = int(input("Enter number: "))
    return(n)

def setUp():
    trey = turtle.Turtle()
    win = turtle.Screen()
    return(win,trey)

def drawLines(t,n):
    for i in range(10):
        t.forward(n)
        t.left(60)

def main():
```



```

n = getInput() #get number of lines to be drawn
w,t = setUp()  #sets up a graphics window and turtle
drawLines(t,n) #repeat 10 times: n steps, turn left 60 degrees

if __name__ == '__main__':
    main()

```

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

```

#Loop through every other letter:
ADDI $sp, $sp, -6    # Set up stack
ADDI $t0, $zero, 65 # Start $t0 at 65 (A)
ADDI $s2, $zero, 75 # Use to test when you reach 75 (K)
SETUP: SB $t0, 0($sp) # Next letter in $t0
ADDI $sp, $sp, 1     # Increment the stack
ADDI $t0, $t0, 2     # Increment the letter
BEQ $t0, $s2, DONE   # Jump to done if $t0 == 75
J SETUP              # If not, jump back to SETUP for loop
DONE: ADDI $t0, $zero, 0 # Null (0) to terminate string
SB $t0, 0($sp)       # Add null to stack
ADDI $sp, $sp, -6    # Set up stack to print
ADDI $v0, $zero, 4   # 4 is for print string
ADDI $a0, $sp, 0     # Set $a0 to stack pointer for printing
syscall              # print to the log

```

**Answer Key:**

ACEGI

- (b) Indicate what modifications are needed to the MIPS program (repeated below) so that it prints out the first 10 upper case letters: ABCDEFGHIJ ?

**Answer Key:** Need to change:

- the first line to have space for 11 characters (ab...j and the null to terminate).
- the corresponding line to allow 11 characters to print (i.e. `ADDI $sp, $sp, -11`).
- increment the register that's holding the letter, `t0`, by a single space (instead of 2)

The resulting program:

```

#Loop through every other letter:
ADDI $sp, $sp, -11    # Set up stack
ADDI $t0, $zero, 65 # Start $t0 at 65 (A)
ADDI $s2, $zero, 75 # Use to test when you reach 75 (K)
SETUP: SB $t0, 0($sp) # Next letter in $t0
ADDI $sp, $sp, 1     # Increment the stack
ADDI $t0, $t0, 1     # Increment the letter
BEQ $t0, $s2, DONE   # Jump to done if $t0 == 75

```

```

J SETUP          # If not, jump back to SETUP for loop
DONE: ADDI $t0, $zero, 0 # Null (0) to terminate string
SB $t0, 0($sp)    # Add null to stack
ADDI $sp, $sp, -11 # Set up stack to print
ADDI $v0, $zero, 4 # 4 is for print string
ADDI $a0, $sp, 0   # Set $a0 to stack pointer for printing
syscall # print to the log

```

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

```

//Lyrics by Lopez & Lopez
#include <iostream>
using namespace std;
int main()
{
(a)   cout << "Let the storm rage ";
      cout << "on\nThe cold never ";
      cout << "bothered me anyway";
      cout << endl;
      return(0);
}

```

#### Answer Key:

Let the storm rage on  
The cold never bothered me anyway

```

//More Elsa
#include <iostream>
using namespace std;
int main()
{
      int count = 2;
      while (count > 0) {
(b)   cout <<"Let it go, ";
      count--;
      }
      cout << "\nThat perfect girl ";
      cout << "is gone\n";
      return(0);
}

```

#### Answer Key:

Let it go, let it go,  
That perfect girl is gone

```

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

```

**Answer Key:**

```

*-*-*
*-*-*
*-*-*
*-*-*
*-*-*

```

10. (a) Translate the following program into a **complete C++ program**:

```

#Python Loops, V3:
for i in range(0,101,2):
    print(i)

```

**Answer Key:**

```

//C++ Loop, V3
#include <iostream>
using namespace std;
int main()
{
    int i;
    for (i = 0; i < 101; i=i+2) {
        cout << i << endl;
    }
    return 0;
}

```

- (b) Write a **complete C++ program** that asks the user for a whole number between -31 and 31 and prints out the number in “two’s complement” notation, using the following algorithm:

- i. Ask the user for a number,  $n$ .
- ii. If the number is negative, print a 1 and let  $x = 32 + n$ .
- iii. If the number is not negative, print a 0 and let  $x = n$ .
- iv. Let  $b = 16$ .
- v. While  $b > 0.5$ :
  - If  $x \geq b$  then print 1, otherwise print 0
  - Let  $x$  be the remainder of dividing  $x$  by  $b$ .
  - Let  $b$  be  $b/2$ .

**Answer Key:**

```
//twos complement
#include <iostream>
using namespace std;
int main()
{
    int n, x, b;
    cout << "Please enter a wole number between -31 and 31: ";
    cin >> n;

    if (n < 0) {
        cout << 1;
        x = 32 + n;
    } else {
        cout << 0;
        x = n;
    }

    b = 16;

    while (b > 0.5) {
        if (x >= b)
            cout << 1;
        else
            cout << 0;
        x = x % b;
        b = b / 2;
    }
    return 0;
}
```