Answer: Answers, inline, preceded by red boxes. See exam for full questions and formatting.

Final Examination, Version 4
CSci 127: Introduction to Computer Science
Hunter College, City University of New York

May 2026

1. (a) Assuming the code is run sequentially, what is the output at each step:

i.
```
stops=["Jct Blvd","90 St","82 St",\
          "74 St","69 St","61 St"]
print(len(stops), "stops")
```

Answer:
```
6 stops
```

ii.
```
print("Last:", stops[-1])
```

Answer:
```
Last: 61 St
```

iii.
```
trs = ["Q72","","","E,F,M,R","",\
       "Q70 SBS"]
jct_tr = trs[0]
print(jct_tr)
```

Answer:
```
Q72
```

iv.
```
nums = [t.count(",") for t in trs]
for s,t,n in zip(stops,trs,nums):
    if len(t) > 0:
        print(s,":",n+1,"transfers.")
```

Answer:
```
Jct Blvd : 1 transfers.
74 St : 4 transfers.
61 St : 1 transfers.
```

(b) Consider the following shell commands:

```
$ ls -l
-rw-r--r--  1 stjohn  staff  19086 Jan 13 16:18 lab8.html
-rw-r--r--  1 stjohn  staff  21510 May  5 14:53 lab9.html
-rw-r--r--  1 stjohn  staff  93628 Apr  1 16:35 ps.html
-rw-r--r--  1 stjohn  staff   4101 Jan 21 07:42 resources.html
-rw-r--r--  1 stjohn  staff  17283 Jan  9 14:59 syl.html
```

Assuming the commands below are run sequentially, what is the output after each has run:

i. `$ ls`

Answer:

```
lab8.html  lab9.html  ps.html  resources.html  syl.html
```

ii. `$ ls -l | grep "Apr" | wc -l`

Answer:

```
1
```

iii.
```
$ echo "Labs:"
$ ls l*.html | wc -l
```

Answer:

```
Labs:
2
```

iv.
```
$ mkdir labs
$ mv l*.html labs
$ ls
```

Answer:

```
labs    ps.html    resources.html    syl.html
```

2. (a) Check **all that apply**:

Answer:

i. What color is the turtle `tess` after this command? `tess.color("#0000DA")`
☐ black ☐ white ☐ red ✓ blue ☐ None of these

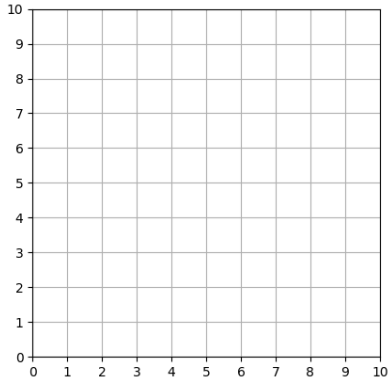
ii. Select all the **binary numbers larger than 8 decimal**:
☐ 0010 ✓ 1101 ☐ 0111 ☐ 0110 ✓ 1110

iii. Select the **odd hexadecimal numbers**:
☐ C ☐ C0 ✓ C7 ✓ 89 ✓ 9F

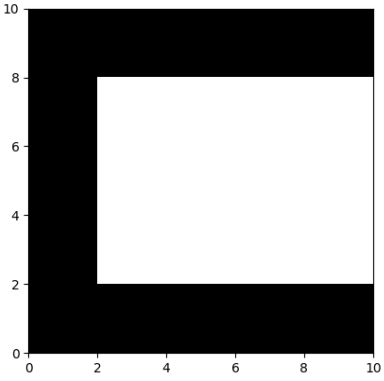
(b) Show the output of this program by shading in the grid:

```
import matplotlib.pyplot as plt
import numpy as np
logo = np.ones( (10,10,3) )
logo[0:2,:,:] = 0
logo[:,0:2,:] = 0
logo[-2:,:,:] = 0
plt.imshow(logo, extent= [0,10,0,10])
plt.show()
```

Output:



Answer:



(c) For each error below, give the line number and the code that would fix the error.

```
1 import pandas as pd
2 irt_data = {'Lines' : [1,2,3,4,5,6,7], 'Colors' : ['R','R','R','G','G','G','P']}
3 print(irt_data items())
4 df = pd.DataFrame(irt_dat)
5 print( df['Colors']] )
```

i.
```
print(irt_data items())
      ^^^^^^^^^^^^^^^^
SyntaxError: invalid syntax.
```

Answer: **Line Number:** 3 **Correct code:** `print(irt_data.items())`

ii.
```
df = pd.DataFrame(irt_dat)
                  ^^^^^^^
NameError: name 'irt_dat' is not defined.
```

Answer: **Line Number:** 4 **Correct code:** `df = pd.DataFrame(irt_data)`

iii.
```
print( df['Colors']] )
                   ^
SyntaxError: closing parenthesis ']' does not match opening parenthesis '('
```

Answer: **Line Number:** 5 **Correct code:** `print( df[['Colors']] )`

3. (a) What will make the following statement true: **Check all that apply.**

```
in1 or not in2
```

Answer:

- ✓ Setting `in1 = False` and `in2 = False`.
- ☐ Setting `in1 = False` and `in2 = True`.
- ☐ All values for `in1` and `in2` make the statement true.
- ✓ Setting `in1 = True` and `in2 = False`.
- ✓ Setting `in1 = True` and `in2 = True`.
- ☐ No values for `in1` and `in2` make the statement true.

(b) What is the value of `out`?

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

Answer:

```
out = False
```

(c) Fill in the values to yield the output:

```
in1 =
in2 =
```

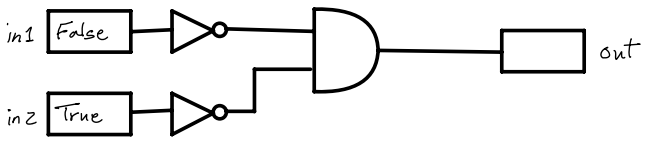
out = False

```
out = (in1 and in2) or (not in1)
```

Answer:

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

(d) What is the output of this circuit?



Answer:

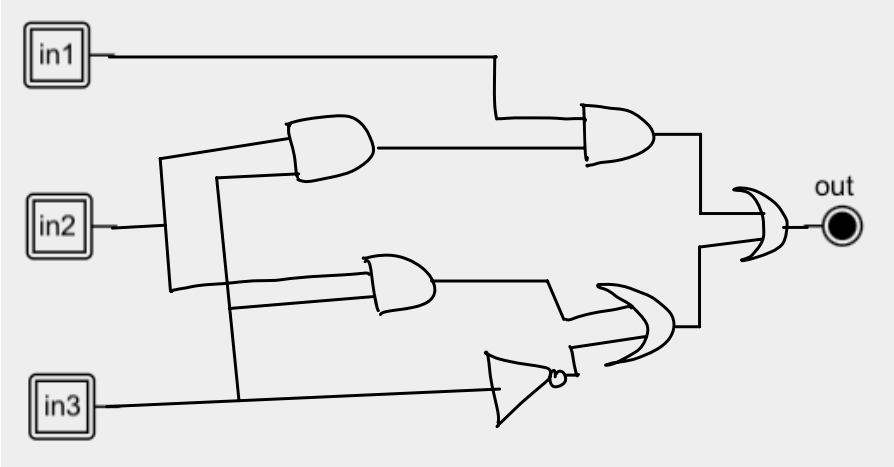
```
out = False
```

(e) Design a circuit that **exactly implements** the logical expression:

```
((in1 and in2) and not in3) or ((in2 or not in3) and in3)
```

Answer:

V3: (in1 and (in2 and in3)) or ((not in2 and in3) or not in3)



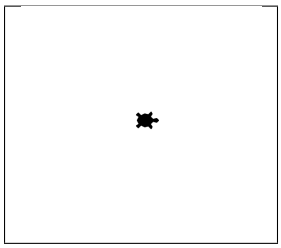
4. (a) Using the turtle and function below, fill in the parameters that yield the output:

```
1  import turtle
2  thea = turtle.Turtle()
3  thea.shape("turtle")
4
5  def ramble(tur, length):
6      if length >= 20:
7          for i in range(4):
8              tur.forward(length)
9              tur.right(360/4)
10         ramble(tur, length/2)
11     elif length < 0:
12         print("Error: neg. length!")
```

i. `ramble( )`

Answer: `thea,` `0` or any value for length between 0 and 19

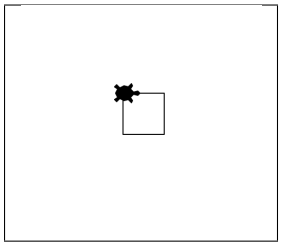
Output:



ii. `ramble( )`

Answer: `thea,` `30` or any value for length between 39 and 20

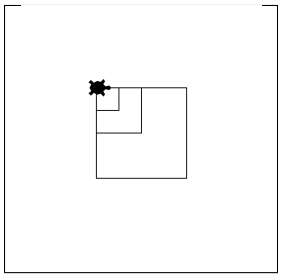
Output:



iii. `ramble( )`

Answer: `thea,` `100` or any number for length between 80 and 159

Output:



(b) What are the formal parameters for the function `ramble()`:

Answer: tur, length

(c) For which values of `num`, will `ramble(thea,num)` print `Error` (check all that apply):

Answer:

- ✓ `num = -10`
- ☐ `num = 0`
- ☐ `num = 10`
- ☐ `num = 100`
- ☐ None of the above.

(d) For which values of `num`, will `ramble(thea,num)` **NOT** execute Line 7 (check all that apply):

Answer:

- ✓ `num = -10`
- ✓ `num = 0`
- ✓ `num = 10`
- ☐ `num = 100`
- ☐ None of the above.

5. Write a function `max_magnitude()` that takes a list of integers and returns the largest absolute value of a number in that list. For example:

```
numbers = [1,-4,55,6,2,4,-90,4,2,3]
print(find_highest(numbers))
```

would print `90` since it has the largest absolute value (e.g. ignore signs) in the inputted list.

Answer:

Libraries:	No additional– just core Python
Input:	a list of integers
Output:	An integer, the largest absolute value

Design Pattern:

Answer:

☐ Accumulator ✓ Max/Min ☐ Finding Duplicates ☐ Searching

Principal Mechanisms (select all that apply):

Answer:

✓ Single Loop ☐ Nested Loop ✓ Conditional (if/else) ☐ Recursion
☐ Indexing/slicing ☐ Dictionary ☐ List Comprehension ☐ Regular Expressions

Process (as a concise and precise LIST OF STEPS / pseudocode):

(Assume libraries have already been imported.)

Answer:

(a) Set up variable `max = 0`.
(b) For each number in the list:
(c) Take the absolute value of the number.
(d) If its larger,
(e) Update `max` with it.

(f) Return `max`.

6. Fill in the Python code below that creates an interactive map using Plotly Express.

Answer:

```
import plotly.express as px
import pandas as pd
#Read in data:
name_str = input('Enter names, separated by spaces: ')
lat_str = input('Enter latitudes, separated by spaces: ')
lon_str = input('Enter longitudes, separated by spaces: ')
#Need to split up the inputted strings into lists:
names = name_str.split(' ')
lats = lat_str.split(' ')
lons = lon_str.split(' ')
#Set up a dictionary of the lists (used to make df):
data = {'latitude': lats, 'longitude': lons, 'name': names}
#Make a DataFrame of the dictionary:
df = pd.DataFrame(data)
#Use column names of df for keyword args:
fig = px.scatter_map(df,
                     lat="latitude",
                     lon="longitude",
                     hover_name="name")
#Save the output:
file_name = input('Enter output file name: ')
fig.write_html(file_name)
```

7. Write a **complete Python program** that

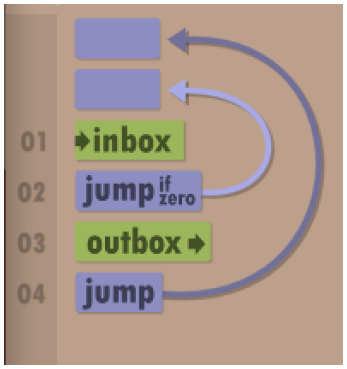
- asks the user for the name of a png file and
- prints the number of pixels that are bright blue (the fraction of blue is above 0.75 and the fraction of green, and the fraction of red are below 0.25).

Answer:

```
#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
countBlue = 0               #Number of pixels that are bright blue

#For every pixel:
for i in range(img.shape[0]):
     for j in range(img.shape[1]):
```

```
            if (img[i,j,2] > 0.75) and (img[i,j,0] < 0.25) and (img[i,j,1] < 0.25):
                countBlue = countBlue + 1

print("Blue count is", countBlue)
```

8. (a) What does the Human Resource Machine (HRM) code output with the following input:



Answer: 5 2 A

Note: if the input is a letter, it is not equal to zero.

(b) Consider the following MIPS program:

```
ADDI $s0, $zero, 20
ADDI $s1, $zero, 10
ADD $s2, $s0, $s1
SUB $s3, $s2, $s1
```

After the program runs, what is the value stored in:

$s0 register	$s1 register	$s2 register	$s3 register
Answer: 20	**Answer:** 10	**Answer:** 30	**Answer:** 20

(c) Consider the MIPS code:

```
1  ADDI $sp, $sp, -7
2  ADDI $t0, $zero, 65
3  ADDI $t1, $zero, 97
4  ADDI $s2, $zero, 3
5  SETUP: SB $t0, 0($sp)
6  ADDI $sp, $sp, 1
7  SB $t1, 0($sp)
8  ADDI $sp, $sp, 1
9  ADDI $s2, $s2, -1
10 BEQ $s2, $zero, DONE
11 J SETUP
12 DONE: ADDI $t0, $zero, 0
13 SB $t0, 0($sp)
14 ADDI $sp, $sp, -6
15 ADDI $v0, $zero, 4
16 ADDI $a0, $sp, 0
17 syscall
```

to change to print the message twice:	* line 4: Set the counter $s2 to 6. * line 14: Set $sp to -12.

Answer:

9. (a) Fill in the missing code to yield the output:

```
//Lyrics by Lopez & Lopez
#include <iostream>
using namespace std;
int main(){
    cout << "It's time to see what ";
    [                                ]
    cout << "limits and break through";
    cout << endl;
    return(0);
}
```

Output:

```
It's time to see what I can do
To test the limits and break through
```

Answer:

```
cout << "I can do\nTo test the ";
```

(b) What is the output:

```
//More Elsa
#include <iostream>
using namespace std;
int main()
{
    int count = 2;
    while (count > 0) {
        cout <<"Let it go, ";
        count--;
    }
    cout << "\nI am one with ";
    cout << "the wind and sky\n";
    return(0);
}
```

Answer:

```
Let it go, Let it go,
I am one with the wind and sky
```

(c) What is the output:

```
#include <iostream>
using namespace std;
int main()
{
    for (int i = 0; i < 5; i++)
    {
        for (int j = 1; j < 5; j++)
        {
            if (i % 2 == 0)
                cout << "+";
            else
                cout << i;
        }
        cout << "?" << endl;
    }
    return 0;
}
```

Answer:

```
++++?
1111?
++++?
3333?
++++?
```

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

Python program:

```
temp=int(input('Enter temperature today:'))
if temp <= 32:
    print('Freezing!')
else:
    celsius = (temp-32)*5/9
    print(celsius, 'degrees')
```

C++ program:

Answer:

```
#include <iostream>
using namespace std;
int main()
{
    int temp;
    float celsius;
    cout << "Enter temperature today: ";
    cin >> temp;

    if (temp <= 32)
        cout << "Freezing!\n";
    else {
        celsius = (temp-32)*5/9;
        cout << celsius << " degrees\n";
    }
    return 0;
}
```

(b) Write a complete C++ program that prints the change in population of the the United States: `p = p + B*p - D*p`, where `p` is the population, `B` is the birth rate of 12.4 births for every 1000 people (12.4/1000) each year, and `D` is the death rate of 8.4 for every 1000 people

(8.4/1000). In 2017, the population of United States was 325.7 million. Your program should ask the user for the number of years and print expected population over those years starting from 2017. Each line should have: the year and the population (in millions). A sample run:

```
Please enter the number of years: 5
Year 2017  325.70
Year 2018  327.00
Year 2019  328.31
Year 2020  329.62
Year 2021  330.94
```

Answer:

```
#include <iostream>
#include <iomanip>
using namespace std;

int main()
{
   float p = 325.7;
   float b = 12.4/1000;
   float d = 8.4/1000;
   float years;

   cout << "Please enter the number of years: ";

   cin >> years;
   for(int i = 2017; i < 2017 + years; i++){
       cout << setprecision(2) << fixed << "Year " << i << "  " << p << endl;
       p = p + ((b*p) - (d*p));
   }
   return 0;
}
```