Answer Key:

FINAL EXAM, VERSION 1
CSci 127: Introduction to Computer Science
Hunter College, City University of New York

20 December 2021

1. (a) Given the quote in the code below, fill in the code to produce the Output on the right:

```
quote = ' "Every moment is a fresh beginning." -T.S Eliot-'
```

  i.
```
print(quote[          ])
```

Answer Key:
```
-10 : -1
```

  ii.
```
print(quote[2:7].              )
```

Answer Key:
```
upper()
```

  iii.
```
print("This quote has", end=" ")
print(quote.count(          )-2, "words.")
```

Answer Key:
```
count(" ")
```

(b) Fill in the code below to produce the Output on the right:
```
numbers = "1, 2, 3, 4, 5"
```
  i.
```
num_list = numbers.              
```

Answer Key:
```
split(', ')
```

  ii.
```
for n in num_list :
    print(          )
```
Answer Key:

```
int(n) + 1
```

(c) Consider the following shell commands:

```
$ ls
bronx.html    logo.png    queens.html snow.png
```

  i. What is the output for:
```
$ mkdir maps images
$ mv *html maps
$ ls
```

Answer Key:

```
images logo.png maps      snow.png
```

  ii. What is the output for:
```
$mv *.png images
$ cd maps
$ ls | grep ee
```

Answer Key:

```
queens.html
```

  iii. What is the output for:
```
$ cd ../
$ ls
```

Answer Key:

```
images maps
```

2. (a) Select the color corresponding to the rgb values below:

Answer Key:

  i. `rgb = (255, 255, 255)`
    ☐ black   ☐ red   **X** white   ☐ gray   ☐ purple

  ii. `rgb = "#AB0000"`
    ☐ black   **X** red   ☐ white   ☐ gray   ☐ purple

    iii. `rgb = (1.0, 0.0, 1.0)`

        ☐ black   ☐ red   ☐ white   ☐ gray   **X** purple

    iv. Select the smallest Hexadecimal number:

        X 0F   ☐ 99   ☐ A0   ☐ FF   ☐ C3

    v. What is the Binary number equivalent to decimal 40?

        ☐ 110100   ☐ 011101   **X** 101000   ☐ 000111   ☐ 101010

(b) Given the list `words` below, fill in the code to produce the Output on the right:

```
words = [ "fast", "clear", "light", "hot", "cold"]
```

i. **Answer Key:**
```
for i in range( 3 ):
print(words[i], end=" ")
```

Output:
```
fast clear light
```

ii. **Answer Key:**
```
for j in range( 1, 5, 3 ):
    print(words[j], end=" ")
```
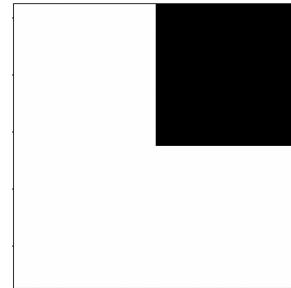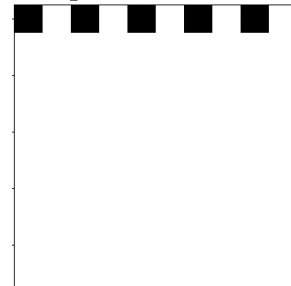
Output:
```
clear cold
```

iii. **Answer Key:**

```
import numpy as np
import matplotlib.pyplot as plt
im = np.ones( (10,10,3) )
im[ : 5 , 5 : , :]  = 0
plt.imshow(im)
plt.show()
```

Output:

iv. **Answer Key:**

```
import numpy as np
import matplotlib.pyplot as plt
im = np.ones( (10,10,3) )
im[ : 1 , : : 2 , :]  = 0
plt.imshow(im)
plt.show()
```

Output:

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

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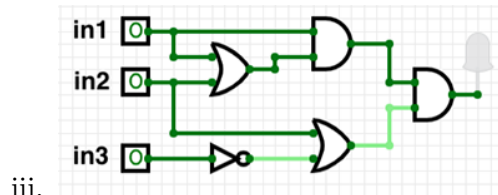
i. `in2 = True`

             `out = not in1 and in2`

Answer Key:

`out = True`

             `in1 = False`

             `in2 = True`

ii.

             `in3 = in1 or not in2`

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

Answer Key:

`out = True`

iii.

             `in1 = True`
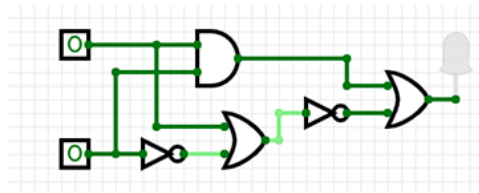
             `in2 = True`

             `in3 = False`

Answer Key:

`out = True`

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

        `(in1 and in2) or not(in1 or not in2)`
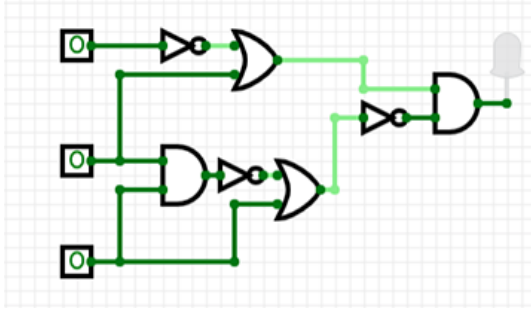
Answer Key:

(c) Fill in the circuit with the gate-symbol or gate-name that implements the logical expression:

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

Answer Key:

4. Consider the following functions:

```
def count_larger(l, n):
  count = 0
  for i in range(len(l)):
    if compare(l[i], n):
        count += 1
  return count
```

```
def compare(num, comp):
    return num > comp


def main():
  numbers = [21, 34, 69, 62, 82, 46, 15]
  print(count_larger(numbers, 50))
```

(a) What are the formal parameters for `compare()`?

Answer Key: num, comp

(b) What are the actual parameters for `count_larger`?

Answer Key: numbers, 50

(c) How many calls are made to `compare()` after calling `main()`?

Answer Key: 7

(d) What is the output after calling `main()`?

Output:

Answer Key:

3

5. Design an algorithm that asks the user for the name of a text file containing a grid of numbers and loads it into a 2D array of integers (think like an image without the color channel), then outputs the index (`row, col`) of the LARGEST number in the array.
Libraries:

Answer Key: numpy
Input:

Answer Key: The input file
Output:

Answer Key: The index of the largest number
Design Pattern:

Answer Key: ☐ Search   ☐ Find Min   **X** Find Max   ☐ Find All
Principal Mechanisms (select all that apply):

Answer Key: ☐ Search   ☐ Single Loop   **X** Nested Loop   **X** Conditional (if/else) statement
XIndexing / Slicing   ☐ `split()`   **X** `input()`

Process (as a concise and precise LIST OF STEPS / pseudocode):
(Assume libraries have already been imported.)

Answer Key:

(a) Ask the user for input file name

(b) Load the data into a numpy array, call it `grid`

(c) Set variables max_row and max_col to 0

(d) Use a nested loop to consider every number in `grid` looping for rows in outer loop and columns in inner loop

    i. if the current number (the number at grid[current_row, current_column] ¿ grid[max_row, max_col], set max_row to current_row and set max_col to current_column

(e) Return max_row and max_col

6. Consider the `open_restaurants.csv` dataset for **restaurant reopening applications** under Phase Two of the New York Forward Plan to place outdoor seating in front of their business on the sidewalk and/or roadway. **Each row in the dataset corresponds to an application**. A snapshot of the data is given in the image below:

Seating Interest	Restaurant Name	Borough	Sidewalk Area	Roadway Area	Approved for Sidewalk Seating	Approved for Roadway Seating
sidewalk	HUNGRY GHOST	Manhattan	200	640	yes	no
both	Prince Laban&Chinese rest.	Queens	144	144	yes	yes
sidewalk	Philly Pretzel Factory	Brooklyn	6500	920	yes	no
both	BICKLES TO GO	Bronx	100	160	yes	yes
roadway	STARBUCKS	Manhattan	160	160	no	yes
roadway	OVENLY	Brooklyn	40	168	no	yes
sidewalk	LE PAIN QUOTIDIEN	Manhattan	105	280	yes	no
both	Le Pain Quotidien GCW	Manhattan	90	240	yes	yes
both	Asian Kabab and Curry	Brooklyn	60	60	yes	yes

Fill in the Python program below:

Answer Key:

```python
#Import the libraries for data frames
import pandas as pd

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

#Read input data into data frame:
df = pd.read_csv(csvFile)

#Print the number of applications for each Seating Interest
# (i.e. number of applications for sidewalk, number for roadway, etc.)
print(df['Seating Interest'].value_counts())

#Group the data by Borough to extract applications in Queens
#use groupby and get_group
queens = df.groupby('Borough').get_group('Queens')

#Print the largest sidewalk area in Queens
print(queens['Sidewalk Area'].max())
```

7. Consider the Python program below to display the multiplication table for an input number. **Fill-in the functions** based on the comments and the overall program. Pay attention to the sample output in the comments in-order to implement the function correctly. Note that the sample output for print_mult_talbe is not complete to save space, **your function must display the full multiplication table.**

Answer Key:

```
def print_mult_table(n):
  # Iterate 10 times from i = 1 to 10
  for i in range(1, 11):
    print(n, 'X', i, '=', n*i)
```

Answer Key:

```
def validate_input(num):
  while(num < 1 or num > 10):
     print('Please enter a number between 1 and 10.')
     num = int(input("Display the multiplication table of? "))
   return num

# Display multiplication table of an input number in range 1 - 10
def main():
  num = int(input("Display multiplication table of? "))
  num = validate_input(num)

  #print the multiplication table of num
  print_mult_table(num)
```

8. (a) What does the MIPS program below print:

   Answer Key:

   ```
   Hello!
   ```

   (b) Modify the program to print out `Hall!`
   Shade in the box for each line or line-pair that needs to be changed and rewrite the instruction
   below. If the line needs to be deleted, write `Delete`.

   Answer Key:

   ```
   # Print Hall!
   ADDI $sp, $sp, -6
   ADDI $t0, $zero, 72 # H
   SB $t0, 0($sp)
   ADDI $t0, $zero, 97 # a
   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, 33 # !
SB $t0, 4($sp)
ADDI $t0, $zero, 0 # (null)
SB $t0, 5($sp)

ADDI $v0, $zero, 4 # 4 is for print string
ADDI $a0, $sp, 0
syscall  # print to the log
```

(c) Modify the MIPS program below to count from 30 to 0, down by 5. Shade in the box for each line that needs to be changed and rewrite the instruction below.

Answer Key:

```
ADDI $s0, $zero, 30 #set s0 to 30
ADDI $s1, $zero, 5 #use to decrement counter, $s0
ADDI $s2, $zero, 0 #use to compare for branching
AGAIN: SUB $s0, $s0, $s1
BEQ $s0, $s2, DONE
J AGAIN
DONE: #To break out of the loop
```

(d) After the modification, how many times is the line labeled `AGAIN:` executed?

Answer Key:

```
6 times.
```

9. Fill in the C++ programs below to produce the Output on the right.

```
#include <iostream>
using namespace std;
int main()
{

    for(int i = 0; i <=30;            ){
```

(a) **Answer Key:**

```
i += 10
```

```
        cout << i*2  << endl;
    }
    return 0;
}
```

```
#include <iostream>
using namespace std;
int main()
{
    int count = 5;
    int num = 2;
```

(b)   while(count ☐ && num ☐){

```
        cout << count << " " << num << endl;
        count -=1;
        if(count % 2 == 0)
            num -=1;
    }
    return 0;
}
```

Answer Key:

```
count > 0 \&\& num >= 0
or
count >= 1 \&\& num > -1
#include <iostream>
using namespace std;
int main(){

    for (int i = 5; ☐ ; i--){
```

(c) **Answer Key:**

```
  i > -5
  or
  i >= -4

        cout << "Still counting!" << endl;
  }
    return 0;
}
```

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

```
for i in range(0,10,2):
  for j in range(i,0,-1):
    print(i, j)
```

Answer Key:

```
#include <iostream>
using namespace std;
int main(){
    for(int i = 0; i < 10; i+=2){
        for(int j=i; j>0; j--){
            cout << i << " " << j << endl;
        }
    }
    return 0;
}
```

(b) Write a **complete C++ program** that asks the user for their age and outputs the age category on a new line as follows:

- "Child" if the user is 18 or younger
- "Adult" if the user is older than 18 but less than 65
- "Senior" otherwise

Answer Key:

```
//include library and namespace
#include <iostream>
using namespace std;

//function signature
int main(){

    //declare variables
    int age;

    //obtain input
    cout << "Please enter your age: ";
    cin >> age;

    //output age category
    if(age <= 18)
        cout << "Child" << endl;
    else if(age < 65)
        cout << "Adult" << endl;
    else
        cout << "Senior" << endl;

    return 0;
}
```