**Answer Key:**

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

20 May 2022

1. (a) Fill in the code below to produce the Output on the right:

```
workdays = "Monday?Tuesday?Wednesday?Thursday?"
summer_months = "*June*July*August*"
long_weekend = "Friday_Saturday_Sunday"
seasons = "+Spring+Summer+Fall+Winter"
```

  i.
```
print(            ],            ])
```

   **Answer Key:**
```
print(long_weekend[-6:], workdays[0:6])
```

  ii.
```
day_list = workdays[          ].split(        )

print("Our week has", len(        ), "days.")
```

   **Answer Key:**
```
day_list = workdays[:-1].split('?')
print("Our week has", len(day_list), "days.")
```

  iii.
```
for day in

    print(            )
```

   **Answer Key:**
```
for day in day_list:
    print(day.upper())
```

   (b) Consider the following shell commands:

```
$ ls
hello.cpp pictures pp_hello.py temp
```

i. What is the output for:
```
$ mv hello.cpp    p1.cpp
$ ls
```

**Answer Key:**

```
p1.cpp pictures pp_hello.py temp
```

ii. What is the output for:
```
$ mkdir c++
$ mv *.cpp   c++
$ ls
```

**Answer Key:**

```
c++  pictures pp_hello.py temp
```

iii. What is the output for:
```
$ cd c++
$ mkdir p50_60
$ mkdir pp_5
$ ls | grep pp
```

**Answer Key:**

```
p1.cpp    pp_5
```

2. (a) Select the correct option.

**Answer Key:**

i. What color is tina? `tina.color(0.0,0.0,0.0)`
   **X** black       ☐ red       ☐ white       ☐ gray       ☐ purple

ii. Select the LARGEST Binary number:
   ☐ 0110       ☐ 1001       **X** 1101       ☐ 1011       ☐ 0000

iii. Select the SMALLEST Hexadecimal number:
   ☐ A0       ☐ 22       **X** 0A       ☐ FF       ☐ CD

iv. What is the Binary number equivalent to decimal 22?
   ☐ 11010       ☐ 01110       **X** 10110       ☐ 00011       ☐ 10101

v. What is the Hexadecimal number equivalent to decimal 20?

**X** 14          ☐ A1          ☐ F0          ☐ 1F          ☐ 18

(b) Fill in the code to produce the Output on the right:

```
nums = [ 23, 45, 76, 23, 98, 45 , 11, 4, 33, 29, 5, 66]
```

i. **Answer Key:**
```
for i in range( 2,5 ):
print(nums[i], end=" ")
```

**Output:**

```
76 23 98
```

ii. **Answer Key:**
```
for j in range( 1,8,2 ):
   print(nums[j], end=" ")
```
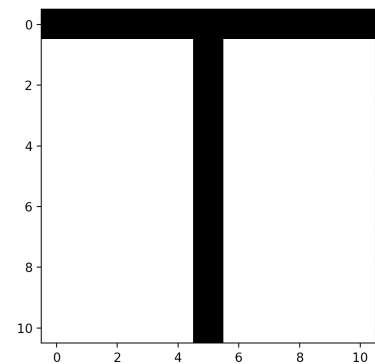
**Output:**

```
45 23 45 4
```

iii. **Answer Key:**

```
import numpy as np
import matplotlib.pyplot as plt
img = np.ones( (11,11,3) )
img[  0  ,  :  , :]  = 0
img[  :  ,  5  , :]  = 0
plt.imshow(im)
plt.show()
```

**Output:**



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

i.
```
in1 = False
in2 = True
out = not (in1 and in2) and (not in1 and in2)
```
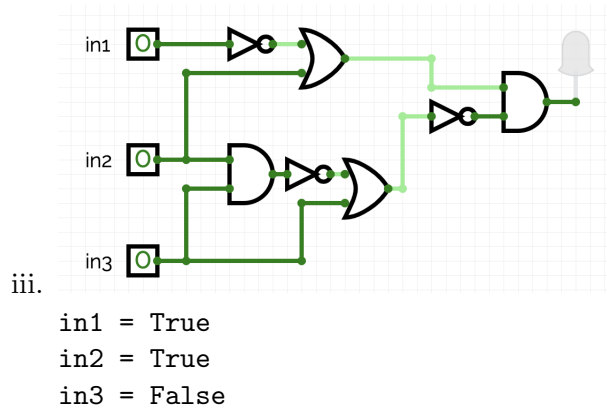
**Answer Key:**
```
out = True
```

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

**Answer Key:**
```
out = False
```

iii.

```
in1 = True
in2 = True
in3 = False
```
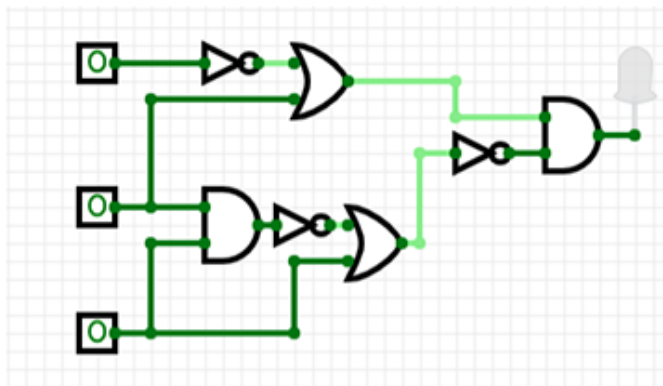
**Answer Key:**

```
out = False
```

(b) Draw a circuit 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 jig(n, m):
  for i in range(n):
    if(i % 3 == 0):
      print(saw(i, m))
```

```
def saw(i, m):
  for j in range(i):
    m+=1
  return m


def main():
    jig(10, 5)
```

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

**Answer Key:** i, m

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

   **Answer Key:** 10, 5

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

   **Answer Key:** 4

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

   **Answer Key:**
   ```
   5
   8
   11
   14
   ```

5. A palindrome is a string (word or sentence, e.g. "madam" or "nurses run") that reads the same backward as forward. Design an algorithm that reads a string and outputs whether it is a palindrome or not. You must write detailed **pseudocode** as a precise list of steps that completely and precisely describe the algorithm.
   **Libraries (if any):**

   **Answer Key:** no libraries necessary
   **Input:**

   **Answer Key:** The input string
   **Output:**

   **Answer Key:** A message saying whether or not the input is a palindrome.

   **Principal Mechanisms (select all that apply):**

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

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

**Answer Key:**

(a) Ask user for input string and store it in variable

(b) Remove spaces from string either by looping through the string and copy character to a new string only if not space or by using the string function `replace(" ", "")`

(c) Make a copy of the string reversed, either by looping through the original string backwards (from str[-1] to str[0]) and append characters to reversed string, or by slicing backwards `reversed = str[-1:  : -1]`

(d) If looping, steps 1 and 2 can be combined (loop backwards and copy character only if not space)

(e) Use if else statement to compare input string to reversed string

(f) If input string is the same as reversed string, output a message saying input string is a palindrome

(g) Else, output a message saying input string is not a palindrome

6. Consider the `art_library.csv` from the **Art Garfunkel's Library** from kaggle. **Each row in the dataset corresponds to a book**. A snapshot of the data is given in the image below:

| Date Read | Author | Books | Year Published | Pages | Favorite |
|---|---|---|---|---|---|
| **Jun-68** | Jean-Jacques Rousseau | The Confessions | 1781 | 606 | 1 |
| **Jun-68** | Erich Fromm | The Art of Loving | 1956 | 146 | 1 |
| **Jun-68** | Mark Twain | The Adventures of Huckleberry Finn | 1884 | 288 | 0 |
| **Jul-68** | James Thurber | My Life and Hard Times | 1933 | 115 | 0 |

■    ■    ■

| Date Read | Author | Books | Year Published | Pages | Favorite |
|---|---|---|---|---|---|
| **Jan-22** | James Michener | Caravans | 1963 | 320 | 0 |
| **Jan-22** | Abraham Lincoln | The Spiritual Growth of a Public Man | 1973 | 47 | 0 |
| **Feb-22** | Joe Scarborough | Saving Freedom | 2020 | 272 | 0 |

Fill in the Python program below:

**Answer Key:**

```
#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:
lib = pd.read_csv(csvFile)

#Print the number of applications for date
print(lib['Date Read'].value_counts())

#Group the data by author to extract books written by Jean-Jacques Rousseau
rousseau = lib.groupby('Author').get_group('Jean-Jacques Rousseau')


#Print the latest year a Rousseau book was published
print(rousseau['Year Published'].max())
```

7. Fill in the following functions that are part of a program that extracts data from a CSV file:

   - **getData()**: asks the user for the name of the CSV and returns a DataFrame of the contents.
   - **extract()**: computes and returns the maximum, minimum and average value of the input column
   - **getList()**: returns a list of length (max-min)/avg, containing equally spaced numbers in range [min, max]

   **Answer Key:**

```
import pandas as pd
def getData():
    """
    Asks the user for the name of the CSV and
    Returns a dataframe of the contents.
    """
    inF = input('Enter CSV file name: ')
    df = pd.read_csv(inF)
    return(df)

def extract(df, col):
    """
    Computes and returns the maximum, minimum and average values
    of the column col in dataframe df
    """
    max = df[col].max()
    min = df[col].min()
    avg = df[col].mean()
    return(max, min, avg)
```

```
def getList(max, min, avg):
    """
     Creates and returns a list of equally spaced numbers in range [min, max].
      The length of the list is (max-min)/avg
     """
    l = []
    length = (max-min)/avg
    space = (max - min)/length
    for i in range(int(length)):
        l.append(min) # or l[i] = min
        min = min + space
   return(l)
```

8. (a) What is printed by the MIPS program below:

**Answer Key:**

`AAAAAAAAAAAAAA`

(b) Modify the program to print out "ACEGIK". Shade in the box for each line that needs to be changed and rewrite the instruction below, or add instructions where necessary.

**Answer Key:**

```
#Loop through characters
 ADDI $sp, $sp, -7       # Set up stack
ADDI $s3, $zero, 1       # Store 1 in a register
ADDI $t0, $zero, 65      # Set $t0 at 65 (A)
ADDI $s2, $zero, 7       # Use to test when you reach 7
SETUP: SB $t0, 0($sp)    # Next letter in $t0
ADDI $sp, $sp, 1         # Increment the stack
ADDI $s3, $s3, 1         # Increment the counter by 1
ADDI $t0, $t0, 2         # Increment the letter (added instruction)
BEQ $s3, $s2, DONE       # Jump to done if $s3 == 7
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
```

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

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

(a)
```
    for(            ; i <=15;            ){

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

**Answer Key:**

```
for( int i = 4; i <=15; i +=2)
#include <iostream>
using namespace std;
int main()
{
     int n=-4, m=10;
```

(b)
```
    while(n+m          ){

        n--;
        m+=2;
        cout << n << " " << m << endl;
    }
    return 0;
}
```

**Answer Key:**

```
while(n+m < 10)
```

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

for ( [                    ] ){

**Answer Key:**

```
 for(int i = 5; i < 10; i++)
```

(c)      for ( [                    ] ){

**Answer Key:**

```
for(int j = 10; j >= i; j--)
            cout << i << i+j << " ";
        }
        cout << endl;
    }
    return 0;
}
```

10. (a) Write a **complete C++ program** that repeatedly asks the user for their age until the age is in range [18, 65], then it outputs the age:

**Answer Key:**

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

int main()
{
    int age = 0;
    do{
        cout << "Please enter your age: ";
        cin >> age;
    }while(age <18 || age >65);

    cout << "Your age is " << age << endl;
    return 0;
}
```

(b) The global population has grown from 1 billion in 1800 at a rate of approximately 1.1% per year.
Write a **complete C++ program** that asks the user for a year after 1800 and returns the global population (in billions) in that year.

**Answer Key:**

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

int main()
{
    float population = 1.0;
    int year = 0, init_year=1800;

    cout << "Please enter a year after 1800 and before 2020: ";
    cin >> year;

    while(init_year < year){
        population += population*0.011;
        init_year +=1;
    }

    cout << "The population in year " << year;
    cout << " was "<< population << " billions."<< endl;
    return 0;
}
```

# SCRATCH PAPER