**Answer Key:**

# Final Exam, Version 3

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

23 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(seasons[1:7],workdays[7:14])

* 1. days = long weekend[

].split( )

print("Our weekend has", len( ),"days.")

## Answer Key:

days = long\_weekend[:].split(’\_’) print("Our weekend has", len(days), "days.")

* 1. for d in

print( )

## Answer Key:

for d in days:

print(d.upper())

1. Consider the following shell commands:

$ pwd

/Users/guest

$ ls

bronx.png circuit.txt nand.txt nyc.png temp

* 1. What is the output for:

$ mkdir logic

$ mv \*txt logic

$ ls

## Answer Key:

bronx.png logic nyc.png temp

* 1. What is the output for:

$ cd logic

$ ls

## Answer Key:

circuit.txt nand.txt

* 1. What is the output for:

$ cd ../temp

$ pwd

## Answer Key:

/Users/guest/temp

1. (a) Select the correct option.

## Answer Key:

1. What color is tina? tina.color(1.0,0.0,1.0)

□ black □ red □ white □ gray **X** purple

1. Select the SMALLEST Binary number:

□ 1011 □ 1101 □ 1111 **X** 1010 □ 1110

1. Select the LARGEST Hexadecimal number:

□ AA □ BA **X** DC □ CC □ CD

1. What is the binary number equivalent to decimal 14?

□ 1011 □ 1101 □ 1111 □ 1010 **X** 1110

1. What is the hexadecimal number equivalent to decimal 170?

**X** AA □ BA □ DC □ CC □ CD

1. Fill in the code to produce the Output on the right:

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

## Answer Key:

for i in range( **3, 10** ): print(nums[i], end=" ")

## Output:

23 98 45 11 4 33 29

* 1. **Answer Key:**

for j in range( **1,12,4** ): print(nums[j], end=" ")

## Output:

45 45 29

**Answer Key:**

import numpy as np

import matplotlib.pyplot as plt img = np.ones( (11,11,3) )

**:**

## 5

**Output:**

img[ , , :] = 0

# black row

**0**

## :

img[ , , :] = 0

# black column

plt.imshow(im) plt.show()

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

in1 = False

i. in2 = False

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

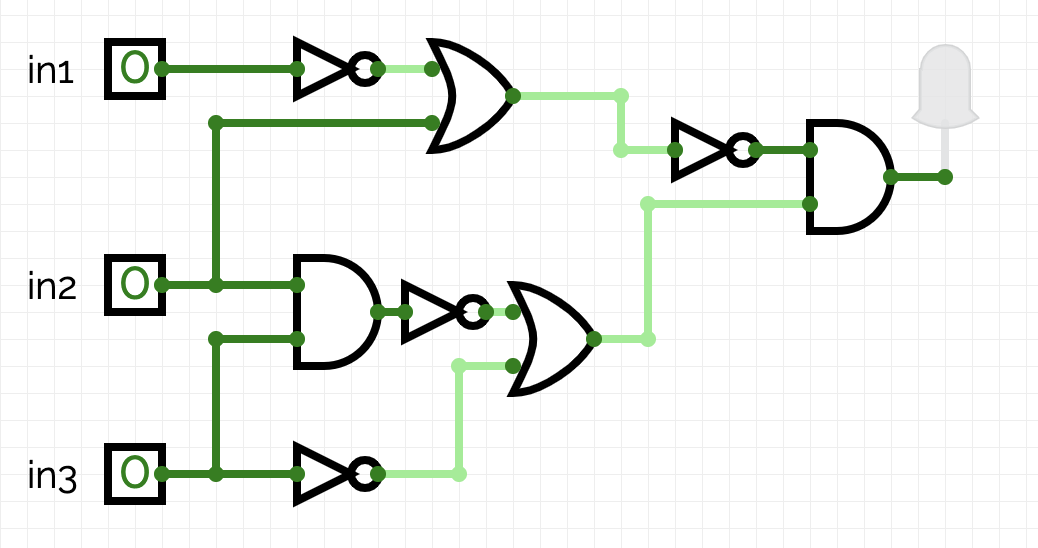
## Answer Key:

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

ii.

## Answer Key:

out = True

iii.

in1 = True in2 = False in3 = True

## Answer Key:

out = True

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

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

## Answer Key:

1. Consider the following functions:

def whoop(n, smile):

for i in range(1,n+1): screech(i, smile) print()

def screech(i, smirk): for j in range(i):

print(smirk, end=’ ’)

def main():

whoop(3, ’^\_^’)

1. What are the formal parameters for screech()?

**Answer Key:** i, smirk

1. What are the actual parameters for whoop()?

## Answer Key:

3, ’^\_^’

1. How many calls are made to screech() after calling main()?

## Answer Key: 3

1. What is the output after calling main()?

## Output:

**Answer Key:**

^\_^

^\_^ ^\_^

^\_^ ^\_^ ^\_^

1. Design an algorithm that asks the user for the name of an image file and the quarter [’TL’, ’TR’, ’BL’, ’BR’] they wish to ”black-out”, where ’TL’ stands for Top Left, ’BL’ stands for Bottom Right and so on. The algorithm then saves a new image where that quarter of the image is black. The name of the new image is ’XXblack.png’ where XX is replaced by one of [’TL’, ’TR’, ’BL’, ’BR’] that the user entered. You must write detailed **pseudocode** as a precise list of steps that completely and precisely describe the algorithm.

## Libraries (if

**any):**

**Answer Key:** pyplot and numpy

## Input:

**Answer Key:** The file name and the quarter

## Output:

**Answer Key:** An image where the corresponding quarter is black

## Principal Mechanisms (select all that apply):

**Answer Key:** □ Search □ 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):

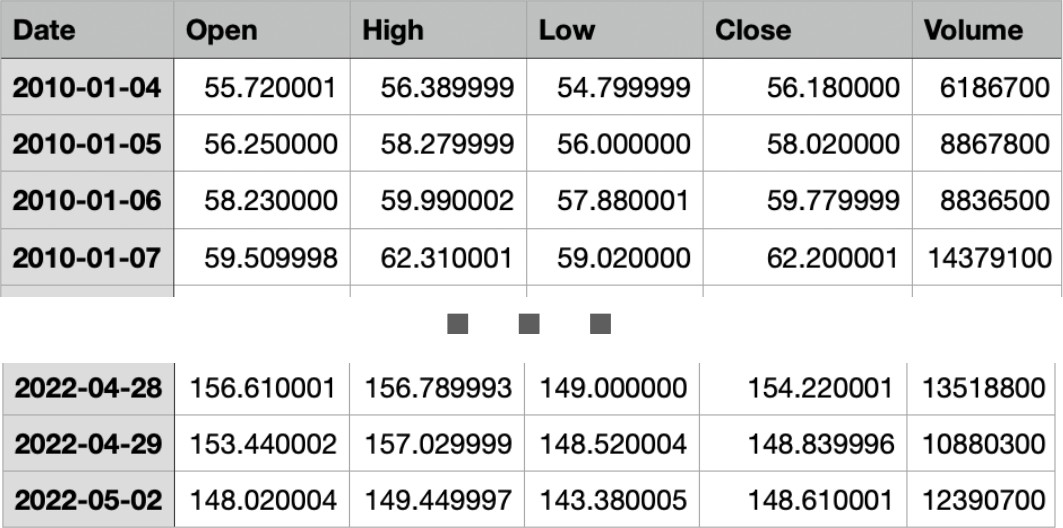
(Assume libraries, if any, have already been imported.)

## Answer Key:

1. Ask the user for the name of an image file
2. Ask the user for the name of a quarter, one of [’TL’, ’TR’, ’BL’, ’BR’]
3. Use pyplot to read the image into a numpy array and give it a name, say img
4. Use img.shape to find the height and width of the image, with height = img.shape[0]

and width = img.shape[1]

1. Use conditionals (if/elif/else statements) to determine which quarter should be black and use slices to set the color of that quarter to black
   1. if quarter == ’TL’, img[ : height//2, : width // 2, : ] = 0
   2. elif quarter == ’BL’, img[ height//2 : , : width // 2, : ] = 0
   3. elif quarter == ’TR’, img[ : height//2, width // 2 : , : ] = 0
   4. else, img[ height//2 : , width // 2 : , : ] = 0
2. use pyplot to save the image to a file with name quarter + ”black.png”, plt.iamsave( quarter + "black.png", img)
3. Consider boeing.csv from the ”Military Stocks during Russia-Ukraine War ” dataset from kaggle, reporting the Boeing Company’s stock prices (in USD $) from January 2010 to May 2022 **Each row in the dataset corresponds to the stock values for one day of trading**. A snapshot of the data is given in the image below:



Fill in the Python program below:

## Answer Key:

#Import the libraries for plotting and data frames import pandas as pd

import matplotlib.pyplot as plt

#Prompt user for input file name:

fin = input("Please enter the name of the Boeing stocks csv file: ")

#Read input data into data frame:

boeing = pd.read\_csv(fin)

#Print the average opening value print(boeing["Open"].mean())

#Print the lowest closing value print( boeing["Close"].min())

#Create a new column called "Range" that computes

#the difference between the highest and lowest value of the stock boeing["Range"] = boeing["High"] - boeing["Low"]

#Plot the newly computed range against the date boeing.plot(x="Date", y="Range")

plt.show()

1. Fill in the following functions that are part of a program that averages the color in an image:

* getData(): asks the user for the name of an image file and returns a numpy array of the pixels
* getAvg(): computes and returns the average (r, g, b) values in img
* avgImg(): returns an image of size rows, cols, with color r, g, b

## Answer Key:

import numpy as np

import matplotlib.pyplot as plt

def getData(): """

Asks the user for the name of an image file Returns a numpy array of the pixels

"""

inF = input(’Enter name of image file ’) img = plt.imread(inF)

return(img)

def getAvg(img):

"""

Computes and returns the average (r, g, b) values in img """

r = img[:,:,0].mean()

g = img[:,:,1].mean()

b = img[:,:,2].mean() return(r, g, b)

def avgImg(rows, cols, r, g, b): """

Creates and returns an image of size rows, cols, with color r, g, b

"""

avg\_img = np.zeros([rows, cols, 3]) avg\_img[:,:,0] = r

avg\_img[:,:,1] = g

avg\_img[:,:,2] = b

return avg\_img

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

## Answer Key:

ZZZZZZZZZ

(b) Modify the program to print out ”ZYXWV”. 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, -6 # Set up stack

ADDI $s3, $zero, 1 # Store 1 in a register ADDI $t0, $zero, 90 # Set $t0 at 90 (Z)

ADDI $s2, $zero, 6 # Use to test when you reach 6 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 BEQ $s3, $s2, DONE # Jump to done if $s3 == 6

ADDI $t0, $t0, -1 # Decrement the letter (added instruction) 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, -5 # 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

1. 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-1 << endl;

}

return 0;

}

## Answer Key:

for( int i = 4; i <=15; i +=2) #include <iostream>

using namespace std; int main()

{

int n=12, m=-5;

1. while(n+m )*{*

cout << n << " " << m << endl; n-=2;

m++;

}

return 0;

}

## Answer Key:

while(n+m > 0)

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

for ( )*{*

## Answer Key:

for(int i = 8; i > 2; i--)

1. for( )*{*

## Answer Key:

for(int j = 0; j <= i; j++)

cout << i << j-i << " ";

}

cout << endl;

}

return 0;

}

1. (a) Write a **complete C++ program** that repeatedly asks the user for two integers until their sum is even, then it outputs the sum:

## Answer Key:

#include <iostream> using namespace std;

int main()

{

int num1=0, num2=0;

do{

cout << "Please enter an integer: ";

cin >> num1;

cout << "Please enter another integer: "; cin >> num2;

}while((num1+num2)%2!=0);

cout << "The sum is " << num1+num2 << endl; return 0;

}

(b) Write a **complete C++ program** that asks the user for an amount and computes the

number of years it takes to triple the amount, if it is subject to an increase of 5% each year.

## Answer Key:

#include <iostream> using namespace std;

int main()

{

float amount = 0.0;

int year = 0;

cout << "Please enter an mount: "; cin >> amount;

int tripled\_amount = amount\*3; while(amount < tripled\_amount){

amount = amount + amount\*0.05; year +=1;

cout << year << " " << amount << endl;

}

cout << "It took " << year << " years to triple your amount to " << amount << end return 0;

}

# SCRATCH PAPER (page left intentionally blank) SCRATCH PAPER (page left intentionally blank)