Columbia College
Vancouver, Canada

Introduction to Computer Science and Programming 1
CSCI120

Sample Midterm

Note: This document has been designed and developed as part of an initiative for creating an OER (Open Education Resource) package for the course CSCI 120 at Columbia College.

Please contact Alireza.davoodi@gmail.com for any comment, modification, and questions.

Terms of use: Please feel free to customize this document as needed

Last Modified: July 2022

Columbia College
Vancouver, Canada

Question	Score
1	
2	
3	
4	
5	
6	
7	
Total (105)	**( ) /105 points**

Problem1: Multiple Choices (15 points)

- Select the answer which is correct the most. Select only one answer

1) What is the output of the following program?

```
def function(numbers):
    num = 0
    for i in range(0, len(numbers)):
        num = num + numbers[i]
    print(num)

function([1,2,3,4,5,6,7,8,9,10])
```

a) No output due to a Compile or Runtime Error
b) 55
c) 45
d) None of the above

2) What is the output of the following program?

```
list = [1,5,3,7,0,8,9,3,2]
counter = 0
sum = 0
while num!=0:
    sum = sum+list[counter]
    counter = counter+1
    num = list[counter]
print (num)
```

a) Error, Compile error
b) 38
c) 16
d) 24

3) What is the output of the following program?

```
if (7 < 0) and (0 <-7):
    print("abhi")
elif (7 > 0) or False:
    print("love")
else:
    print("Hi")
```

a) Hi
b) love
c) abhi
d) Compile or runtime error

4) What is the output of the following program?

```
nameList = ['Harsh', 'Pratik', 'Bob', 'Dhruv']
print("%s %s" % (nameList[1][1], nameList[1][0]))
```

a) r P
b) r r
c) P r
d) Compile or runtime error

5) What is the output of the following program?

```
i = 1
while True:
    if i % 3 == 0:
        break. #comment: the break will terminates the while loop
    i = i + 1
print(i)
```

a) 3
b) 1
c) 4
d) None

Problem2: True/False (10 points)

- **Next to each question add a T or F** depending on whether you think the statement is True or False.

- **True / False:** Based on Demorgan Law: Not (Not A and Not B) = Not A and Not B
- **True / False:** A Python function must have at least one return statement.
- **True / False:** In Python, you can define a variable without specifying its data type
- **True / False:** In Python, to access an element of the set, you need to specify its index number.
- **True / False:** In Python a list is the same a set just no item can be replicated (repeated) in set but in list can have repeated items.

Problem3: Flowchart (10 points)

- Draw a flowchart for the following code that does exactly the same thing.
- Guess what the code does and explain it with an example.

```
number = int(input("Enter a number: "))
if number>0:
    factor = int(input("Enter a factor: "))
    lower = 2
    upper = 10
    if factor<lower or factor>upper:
        print("End")
    else:
        output = ""
        temp1 = number
        while temp1!=0:
            temp1 = number // factor
            temp2 = number % factor
            output = str(temp2) + output
            number = temp1
        print(output)
else:
    print("End")
```

-

Problem4: Coding (15 points)

- The following code checks whether an IP address is a valid address or not. An IP address is valid if it meets all the following requirements:

-
- Rule1: It has only digits.
- Rule2: The maximum length of the address is 16 and the minimum length is 14.
- Rule3: The IP address has 2 parts. The first part is the IP and the second part is port number. The IP and port are separated using a colon :
- Rule4: The IP part contains exactly 12 digits in 4 groups of 3 digits separated using a dot.
- Rule5: The port part of the IP address is at least 2 digits and at most 4 digits and it contains all numbers.
- Look at the examples of valid and invalid IP addresses
    - For instance:
        - 123.456.123.123:8080 is valid
        - 123-456-123-123:8080 is invalid
        - 123456123123:8080 is invalid
        - 123.456.123.123-8080 is invalid
        - 13.46.123.123:8080 is invalid

- Design and implement function which has an IP address as its parameter and checks whether it is a valid IP address or not.
    -

Problem5: Coding (20 points)

Answer the following questions:

Question1: Convert the following code to use a for-loop instead of while-loop to generate exact same results:

```
def function():
    word = input("Enter a word [enter exit to terminate]")
    count = 0
    while word!="exit" and count<10:
        print(word)
        count = count + 1
    return
```

Question2: The following snippet of code receives an input and then checks whether the input is a number. The issue with the following code is that if the user enters a negative number (like -10), the code would not work. Make some changes in this code to also work with negative numbers as well as positive numbers.

```
def function():
    number = input ("enter a number: ")
    if number.isdigit():
        return True
    else:
        return False
```

Question3: The following snippet of code, remove vowels from a string (word) and prints the result, which is equivalent to the word without its vowel letters. Do you think this function will work as expected? If yes please explain and if no, please fix the issue(s) to generate the expected

answer.

```
word = ["h","o","p","e"," ","t", "o", " ","s", "e", "e", " ","y", "o", "u", " ","a", "l", "l"," ","i", "n", " ","t","h","e","
","c","o","l","l","e","g","e"," ","s","o","o","n"]

for i in range(len(word)):
        letter = word[i]
        if letter == "a" or letter == "e" or letter == "i" or letter == "o" or letter == "u":
                word.remove(letter)

print("After removing the vowels the result is %s" %(word))
```

Question4:

- The following python code would not run and gives some compiles errors.
- Look at the code and detect what the errors are and fix them.
- After you fixed it, what is the output of this program?

```
sideLength = 4
def main():
        sideLength = 5
        volume = cubeVolume(sideLength)
        print("The volume is: %d " %volume)

main()

def cubeVolume():
        volume = 3**sideLength
        print(volume)
```

Problem6: Coding (15 points)

- A palindrome list is a list which is the same as its reversed. Write a Python function which take a list of numbers as an input and return True if the list is palindrome or False if the list is not palindrome. (You are allowed to use the methods and functions covered in the lecture but not the methods or functions not covered in the lectures.)
- Example: [] is a palindrome list
- Example: [1,2,1] is a palindrome list
- Example: [1,1,1] is a palindrome list
- Example:[1,2,2] is Not a palindrome list

Question7-Answer

```
def main():
        testIsPalindrome ()

main()
```

```
def isPalindrome (list):
        #complete the function
```

```
def testIsPalindrome ():
    samplelist = [1,2,3,3,2,1]
    #complete the function as a test function (“actualResult”, “expectedResult”)
```

Problem7: Coding (15 points)

- Write a Python function which has a list of points as its input parameter. Each point represents a position in the 3D coordination system (space) and represented as using 3 properties: x, y and z for instance (x=1, y=2, z=3). The function should find and return the point in the input list which has the minimum distance to the center of the coordination system. The center of the coordination system is (0,0,0) in other word for the center, x=0, y=0, z=0

- The distance between two points (x1,y1,z1) and (x2, y2, z2) in coordination system is calculated using the following formula:

  distance = Sqrt ((x1-x2)**2 + (y1-y2)**2 + (z1-z2)**2)
  for example:
  point1=(2,5,6) and point2 = (1,1,1) then the distance is
  distance = sqrt ((2-1)**2 + (5-1)**2 + (6-1)**2) = sqrt (1+16+25) = sqrt (42)

- Completes the following functions:

Question8-Answer

```
def main():
    testFindClosetPointToCenter()
main()
```

```
def testFindClosetPointToCenter():
        #samplePointList = (1,4,1), (3,7,10), (-1,3,4), (4,4,2), (7,-4,0), (10,4,1)
        #Complete code here as a test function (write code to find the closest points from the above points
#       that is closet to the center of the coordination system.
```

```
# This function calculated the distance between point1 and point2. You need to call this function in the above
#function appropriately.
#
def distance(point1, point2):
        dist = 0
        #complete the code here
        return dist
```

```
#points: is a list of points
def findClosetPointToCenter(points):
    #center = (0,0,0)

    #complete code here. You need to come up with an algorithm and use the distance function defined above to find the closet point from the points list to the center
```

Good Luck ☺