ALTERNATE ASSESSMENT QUESTION PAPER: Summer Semester 2021

Module Code: CS4051NI

Module Title: Fundamentals of Computing

Module Leader: Sukrit Shakya (Islington College)

Date: 21 September 2021

Day / Evening: Day

Start Time: 9 AM

Duration: 6 Hours

Test Type: Unseen Examination

Materials supplied:

Materials permitted: Writing equipment only

Warning: Candidates are warned that possession of unauthorized materials in a test is a serious assessment offence.

Instructions to candidates:

This test accounts for **40%** of your total module grades.

You are to **return this test paper**, BEFORE you leave the testing room.

DO NOT TURN PAGE OVER UNTIL INSTRUCTED

Last name = Shrestha

First name = Kishor

London id = 20048913

Instructions:

1. You have 6 hours to complete all the questions.
2. You must answer all the questions.
3. Compile all the answers in a report
4. For questions asking for

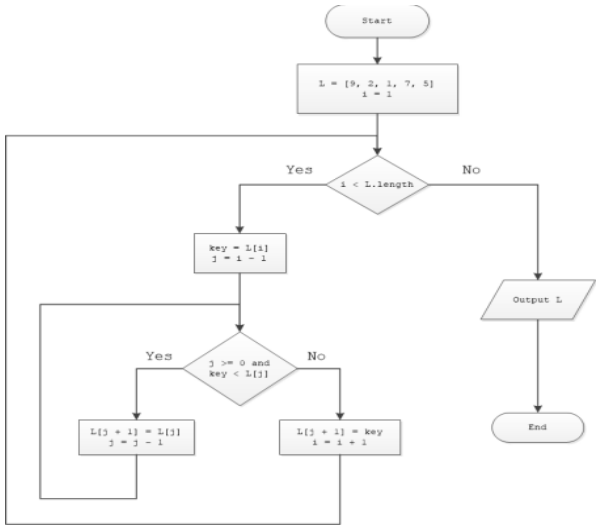
   a .py file, you will need to submit a separate py file for each question. Make sure to name the files according to the question number. a. In the report, you will also need to provide screenshots of the program being executed.

   b. Remember to include screenshots of the source code as well.

Question 1:

You have been provided a flowchart of a sorting algorithm that sorts a list of numbers in ascending order. Given a list, L = [9, 2, 1, 7, 5], the program should sort the list resulting in the following output, L = [1, 2, 5, 7, 9] .

To create the program, you need to analyze the flowchart very carefully and write the Python code. If everything is done correctly, you should obtain the result outlined above. You will need to submit your .py file alongside your document. **[20 Marks]**



Solution

Corrected code:

Question no 1.py - C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 1.py (3.9.6)

File Edit Format Run Options Window Help

```
arr = [9,2,1,7,5];
temp = 0;

#From top to bottom, arrange the array.
for i in range(0, len(arr)):
    for j in range(i+1, len(arr)):
        if(arr[i] > arr[j]):
            temp = arr[i];
            arr[i] = arr[j];
            arr[j] = temp;

print();

#Items of the array will be displayed after sorting.

print("Elements of array sorted in ascending order: ");
for i in range(0, len(arr)):
    print(arr[i], end=" ");
```

Output of corrected code:

IDLE Shell 3.9.6

File Edit Shell Debug Options Window Help

```
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 1.py ===

Elements of array sorted in ascending order:
1 2 5 7 9
>>>
```

Question 2.

You have been provided Python Code for a program below. The program performs operations on a 2D list to change the values. The result of the program has been shown in a screenshot below.

You need to:

1. Analyze the code and explain what's happening. **[10 Marks]**
2. Create a flowchart that describes the operations. The start and endpoints of the program for the flowchart have been outlined in form of comments. **[10 Marks]**

Output:

```
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
--------------
[1, -1, -1]
[1, 5, -1]
[1, 1, 9]
>>>
```

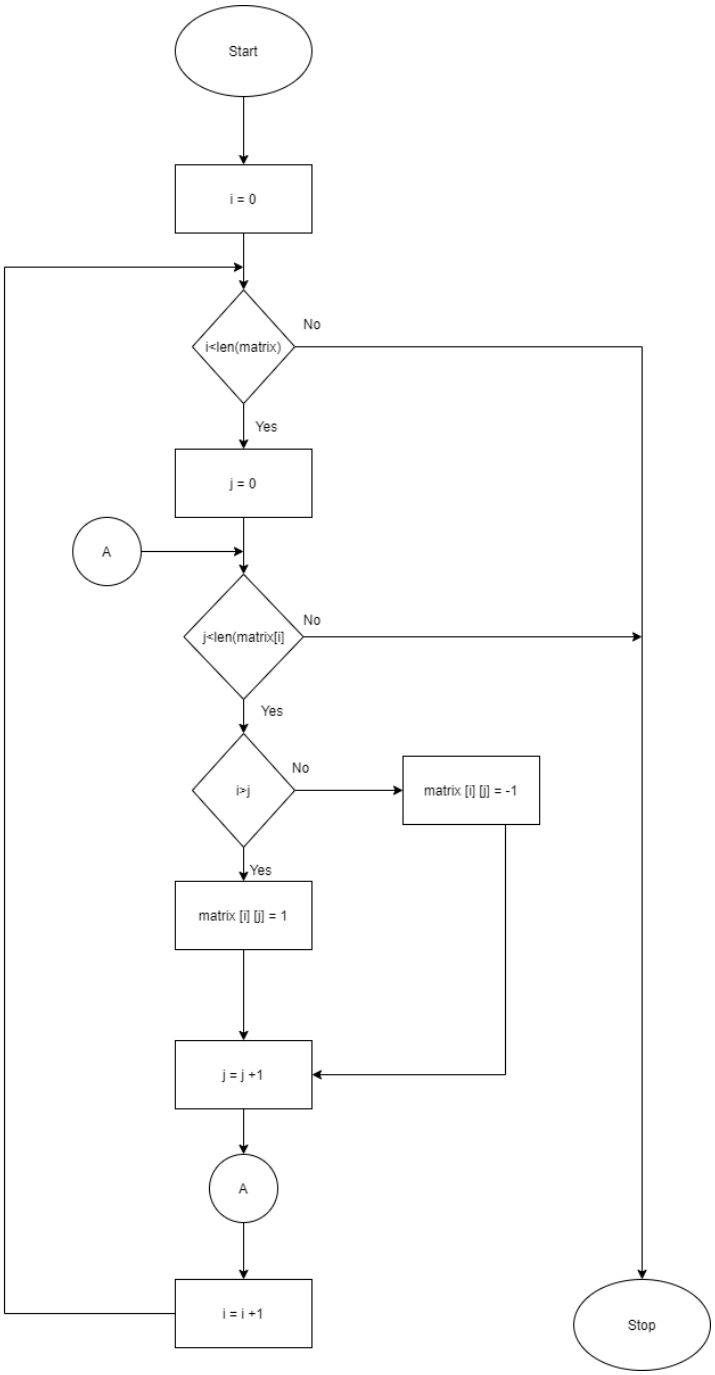
```
matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

for row in matrix:
 print(row)

# start here

i = 0
while i < len(matrix):
 j = 0
 while j < len(matrix[i]):
 if i > j:
 matrix[i][j] = 1
 elif i < j:
 matrix[i][j] = -1
 j = j + 1
 i = i + 1

# stop here

print('--------------')


for row in matrix:
 print(row)
```

Solution

1 Explanation:

In this programme, the matrix is originally started with a values of 3 rows. The for loop has been used to display the column of the initialization matrix, with the size of the row being utilized to execute the loop. It was set to 0 in this case. Meanwhile loop is being used where the length of the matrix is checked, then j is initialized to 0 within the matrix, and another while loop is applied throughout the previous while loop. In this while loop, j is compared to the size of matrix[i], as well as inside this while loop, the if and elif conditions are used to compared I with j. Here, I is compared to see if it is smaller than j, and whether it is, the if loop is executed, and matrix[i][j] is initialized to 1. Matrix[i][j] is initialized to -1 if I is smaller than j. When the if as well as elif loops are finished, j is incremented by 1 just at end of of that while loop for j, and I is expanded by 1 at the finish of the while loop for I and the resultant matrix is printed using the for loop.

2 Flowchart:



Question 3 (a)

Fix the error(s) in the program to obtain the output as shown in the picture below. Write a list of errors you fix in the program.

You will also need to submit your .py file of the corrected code alongside your document. **[10 Marks]**

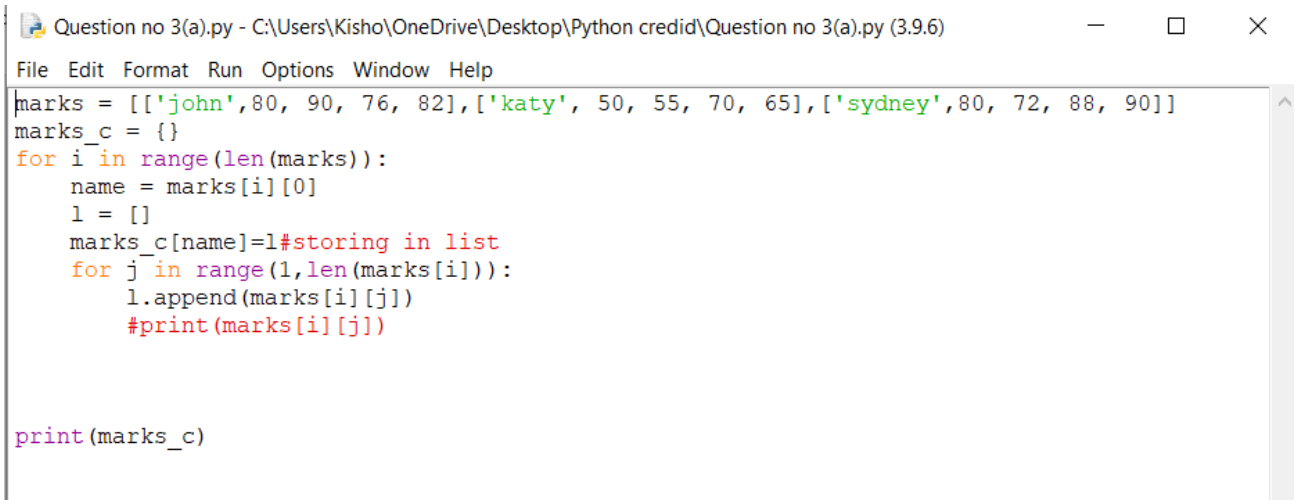
```
{'john': [80, 90, 76, 82], 'katy': [50, 55, 70, 65], 'sydney': [80, 72, 88, 90]}
>>>
```

```
marks = [['john',80, 90, 76, 82],['katy', 50, 55, 70, 65],['sydney',80,
72, 88, 90]]

marks_c = {}
for i in range(len(marks)):
 name = marks[i][0]
 l = []
 for j in range(2,len(marks[i])):
 print(marks[j][i])
 print(marks_c)


print(marks_c)
```

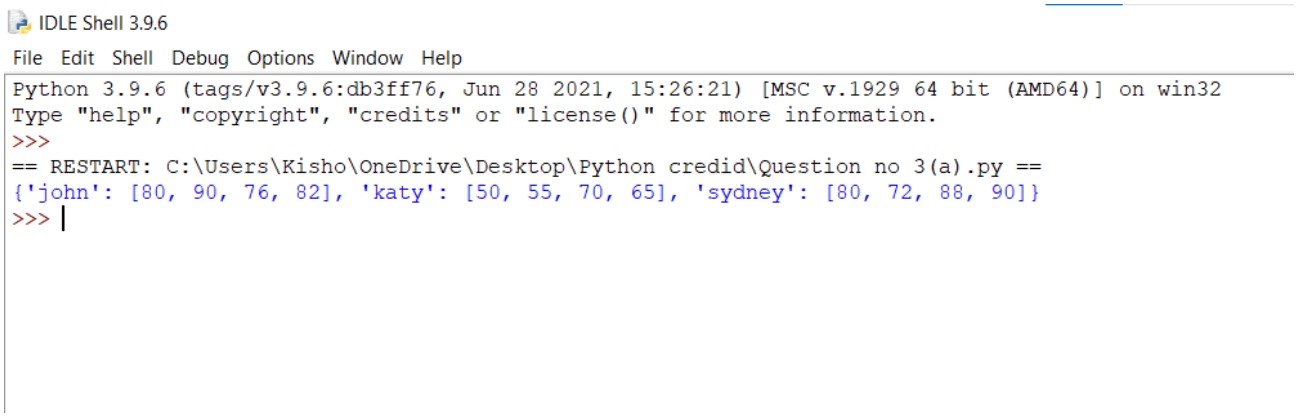
Solution:

Here in line 6 there is a error in, for j in range(2,len(marks[i])): there should be 1 in place of 2 and there should be, 1.append(marks[i] [j]) in second last line. After making complete correction from the code i have represented screenshot of code and output of the code.

Corrected code:



Output of corrected code:



Question 3(b)

Fix the error(s) in the program to obtain the output as shown in the picture below. Write a list of errors you fix in the program.

You will also need to submit your .py file of the corrected code alongside your document. **[10 Marks]**

Expected Output:

```
HI! My name is Ram and I live at Kamalpokhari
9812121212
HI! My name is Rabi and I live at Gaushala
HI! My name is Rabi and I live at Koteshwor
>>>
```

```
class Person:
 def __init__(self, name, address, number)
 """Constructor of Person"""
 self.name == name
 self.address == address
 self.number == number

 def get_Person(self)
 """Returns the name of a person"""
 return name

 def get_all_detail(self)
 """Returns all the details of person"""
 return "HI! My name is " + name + " and I live at " + address

 def get_contact(self)
 """Returns the contact of the person"""
 return number
```

```
class Employee:
 def __init__(self, name, address, number, salary, department):
"""Constructor of Employee"""
 Person.__init__(self, name, address)
 self.salary == salary
 self.department == department

 def get_contact(self):
 """Checks if the number is none and returns the contact of the
person"""
 if number = None:
 return "Does not have a number"
 else:
 return number

 def change_number(self, new_number):
 """method for CHANGING the number of the person"""
self.number += new_number

 def change_address(self, addr):
 """method for CHANGING the address of the person """
self.addr == addr

foo = Person("Ram", "Kamalpokhari",
"9812121212") print(foo.get_all_details())
print(foo.get_contact())
bar = Employee("Rabi", "Gaushala", "9836699636", "40000", "HR")
print(bar.get_all_details())
bar.change_address("Koteshwor")
print(bar.get_all_details()
```

Solution

In program there should be, return self.number in code line 21,There should be def change_address(self,address) in line 25. After making complete correction from the code i have represented screenshot of code and output of the code.

Corrected code:

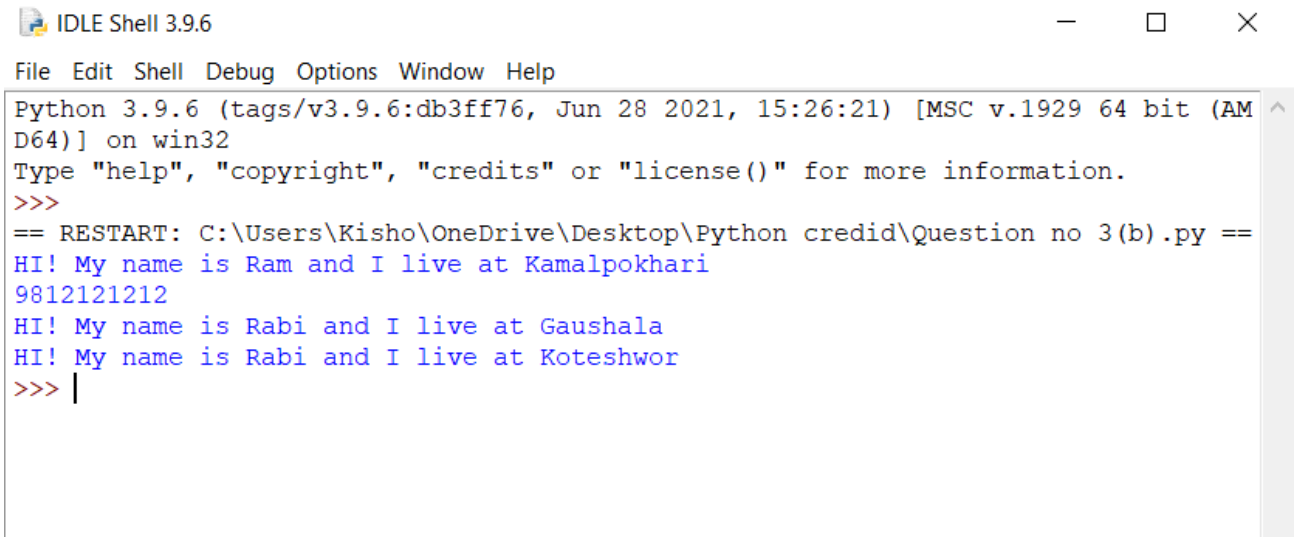
Question no 3(b).py - C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 3(b).py (3.9.6)

File Edit Format Run Options Window Help

```
class Person:
    def __init__(self, name, address, number):
        """Constructor of Person"""
        self.name = name
        self.address = address
        self.number = number

    def get_Person(self):
        """Returns the name of a person"""
        return self.name

    def get_all_detail(self):
        """Returns all the details of person"""
        return "HI! My name is " + self.name + " and I live at " + self.address

    def get_contact(self):
        """Returns the contact of the person"""
        return self.number


class Employee(Person):
    def __init__(self, name, address, number, salary, department):
        """Constructor of Employee"""
        Person.__init__(self, name, address, number)
        self.salary = salary
        self.department = department

    def get_contact(self):
        """Checks if the number is none and returns the contact of the
        person"""
        if self.number == None:
            return "Does not have a number"
        else:
            return self.number

    def change_number(self, new_number):
        """method for CHANGING the number of the person"""
        self.number = new_number

    def change_address(self, addr):
        """method for CHANGING the address of the person """
        self.address = addr
```

```
foo = Person("Ram", "Kamalpokhari", "9812121212")
print(foo.get_all_detail())
print(foo.get_contact())
bar = Employee("Rabi", "Gaushala", "9836699636", "40000", "HR")
print(bar.get_all_detail())
bar.change_address("Koteshwor")
print(bar.get_all_detail())
```

Output of corrected code:



Question 4: Data structures [20 marks]

Python gives you a number of alternative options for storing data collectively. Describe in short all the collection data types in python explaining their workings and uses. Also, give some examples of where they can be used efficiently.

Below is a table consisting of some data about some movies. Chose a suitable collection data type for representing and storing that data in a python program and justify your choice in terms of convenience, economy, and efficiency with examples.

Movie ID	Movie Name	Rent Price	Quantity
M001	Pulp Fiction	$5	30
M002	Lord Of The Rings	$2.5	10
M003	The Revenant	$4	20

Introduction

Python is a high-level programming language enabling text processing, system management, and internet-related activities. Its basic language is relatively short and simple to understand, despite many comparable languages, so it allows the insertion of

modules to do a practically unlimited range of jobs. Python is a real object-oriented programming language that runs on a broad range of systems. There's also a python interpreter developed completely in Java, bolstering python's status as a top web issue solver. Python is mainly used for storing huge amount of data that is mainly needed for future reference. In Python, each value has a data type. In Python programming, everything is an object, hence data types are actual classes, while variables are instances (objects) of such classes. Python has a number of different data types. The following are some of the more important types.

Data Types in Python

- List

  List is a flexible data type that is only available in Python. In some ways, it is indeed similar to an array in C/C++. The collection in Python, on the other hand, has the advantage of being able to contain many types of data at the same time.
  A list is a proper term for an organized series of data expressed in square brackets ([]) and separated by commas (,)

- Tuple

  Tuples, like lists, are ordered sequences of objects. Tuples vary solely in because they are permanent. Tuples can't be changed after they've been formed. Tuples are utilized to start writing data and, since they do not alter constantly, are generally quicker than lists. It is written within parenthesis (), with commas between the components.

- Strings

  A series of characters is referred to as a string. Unicode characters are supported by Python. Separate or even double quotes are being used to describe strings in most cases.

- Set

  A set is an unsorted collection of one-of-a-kind objects. Values separated by commas in brackets specify a set. The set of items are not sorted in any particular sequence.
  a = {2, 4, 4, 5, 3}

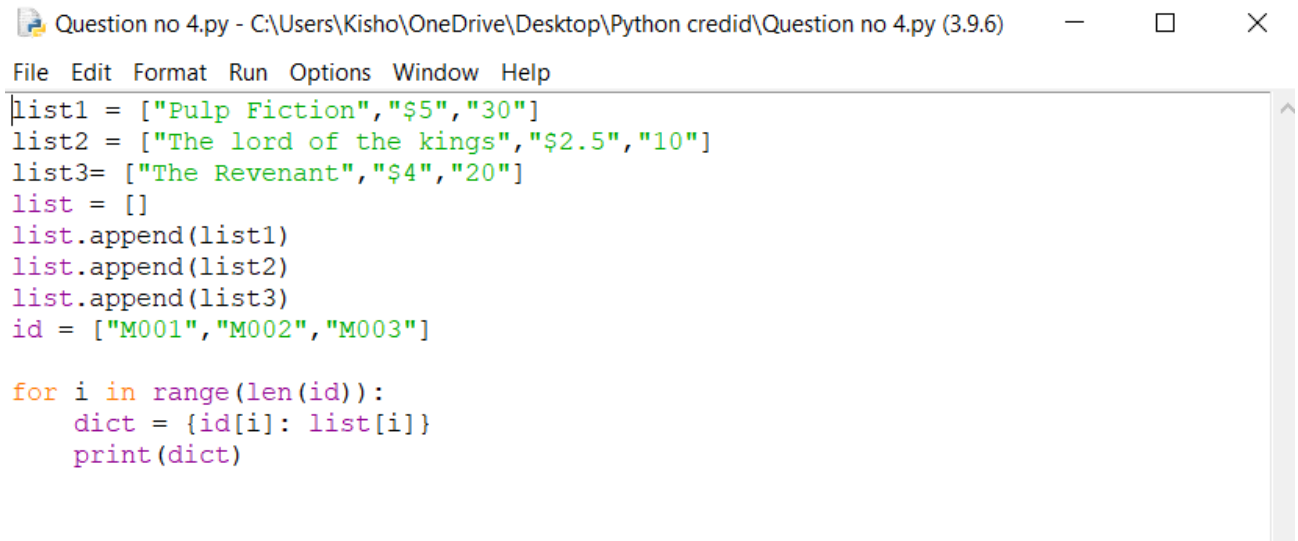
- Dictionary

A dictionary is a set of important combinations that are not in any particular sequence.

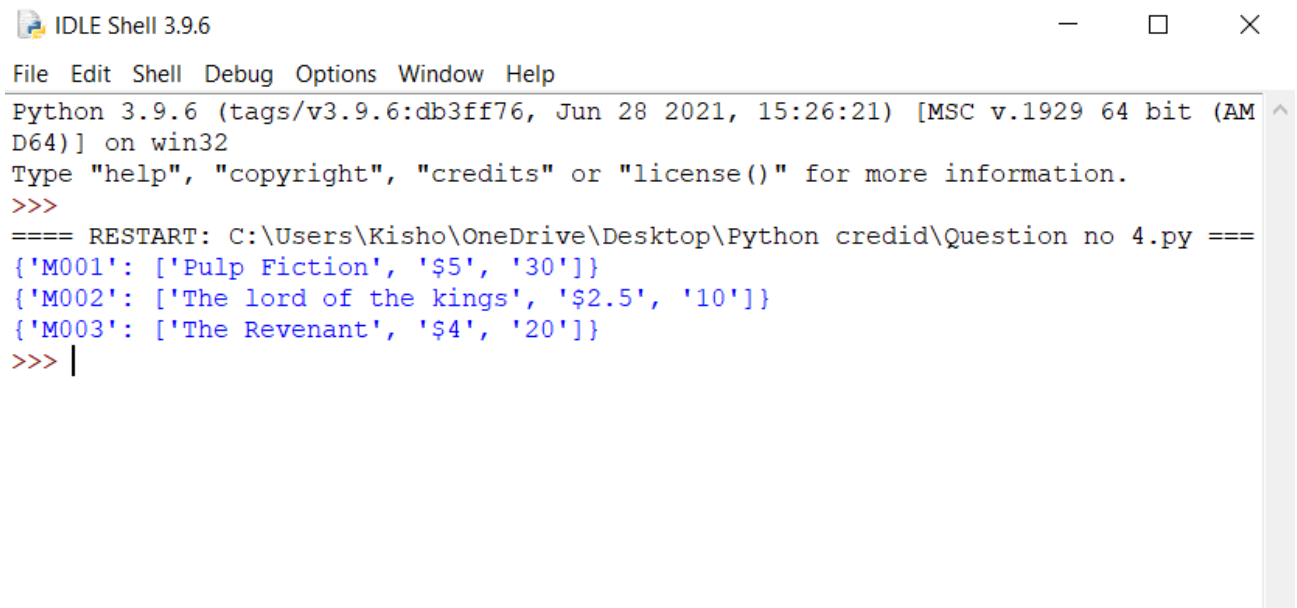
Whenever we obtain a massive volumes of information, we usually use it. The retrieval of data is improved in dictionaries. To get the value, we need to understand the key.

Curly brackets are used to create dictionaries in Python, with each item being such a pair in the form.

Screenshot of Code



Screenshot of output of code



Question 5(a).

Write a program that takes input from the user a string of numbers (eg. “24453”), then all the numbers from the string must be put in a list. The program then computes the sum, maximum and minimum from the list of numbers.

You will need to submit your .py file alongside your document. **[10 Marks]**

Solution

Corrected code:

Question no 5(a).py - C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 5(a).py (3.9.6)

File Edit Format Run Options Window Help

```
number = input(" write  number: ")
def sum_digit(number):
    return sum([int(a) for a in number])

def max_digit(number):
    return max([int(a) for a in number])
def min_digit(number):
    return min ([int(a)for a in number])
print(sum_digit(number))
print(max_digit(number))
print(min_digit(number))
```

Output of corrected code:

IDLE Shell 3.9.6

File Edit Shell Debug Options Window Help

```
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 5(a).py ==
 write  number: 24453
18
5
2
>>>
```

Question 5(b).

Write a program that takes 2 words as input from the user and prints out a list containing the letters which the 2 words have in common. For example, if word1 is “english” and word2 is “nepali”, the output should be [“l”, “e”, “i”, “n”].

You will need to submit your .py file alongside your document. **[10 Marks]**

Solution

Corrected code:

Question no 5(b).py - C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 5(b).py (3.9.6)

File Edit Format Run Options Window Help

```
word1 = input("Enter first word: ").lower()
word2 = input("Enter second word: ").lower()

commonLetter = (set(word1)).intersection(set(word2))

print("Common letters are:", list(commonLetter))
```

Output of corrected code:

IDLE Shell 3.9.6

File Edit Shell Debug Options Window Help

```
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\Kisho\OneDrive\Desktop\Python credid\Question no 5(b).py ==
Enter first word: english
Enter second word: nepali
Common letters are: ['n', 'e', 'i', 'l']
>>>
```

Marking Breakdown for Assessment	
Criteria	**Total Marks**
1. Question 1	
1.1 Screenshots	*3*
1.2 Python Program	
1.2.1 Correctness & Implementation	*15*
1.2.2 Comments	*2*
2. Question 2	
2.1 Code ex++++planation and analysis	*10*
2.2 Flowchart	*10*
3. Question 3 (a): Correctness + Screenshot	*10*
4. Question 3 (b): Correctness + Screenshot	*10*
5. Question 4	*20*
6. Question 5 (a): Correctness + Screenshot	*10*
7. Question 5 (b): Correctness + Screenshot	*10*
Total:	**100**