*Fall 2021*

**CSC/BIF 243**

**Introduction to Object Oriented Programming**

*Course Instructor: Joe Khalife*

LAB06

**Problem 1**

Evaluate manually the following expressions. **Indicate the type of the result**.

|  |  |  |
| --- | --- | --- |
| **Expression** | **Value** | **Type** |
| 12 / 3 + 5 \* 2 - 5 |  |  |
| (12 + 3 ) / / 2 \* 5 |  |  |
| (15 // 4 != 15 % 4) or (‘A’ == ‘a’) |  |  |

Given:

title = "Hello World"

L = [ 1, 2, [‘yes’, 4], ‘No’]

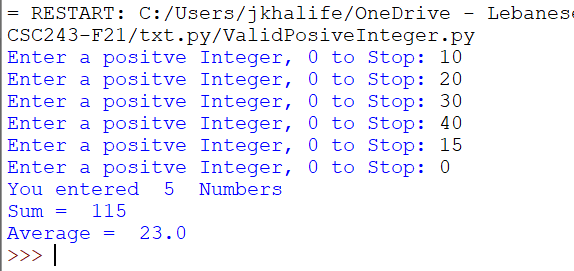
Evaluate the following expressions. **Indicate the type of the result**.

|  |  |  |
| --- | --- | --- |
| **Expression** | **Value** | **Type** |
| len(L) |  |  |
| title[2:6] |  |  |
| L[2:6] |  |  |
| title.index(‘l’) |  |  |
| title.index(‘e’,2) |  |  |
| L[2][1] |  |  |
| ‘yes’ in L |  |  |

**Check your answers using idle and describe any difference.**

**Problem 2**

Write a program that read positive numbers until 0 is entered and output count, sum and average of the numbers entered:

****

**Problem 3**

Write a program that randomly generates an integer between 0 and 100, inclusive. The program prompts the user to enter a number continuously until the number matches the randomly generated number. For each user input, the program tells the user whether the input is too low or too high, so the user can choose the next input intelligently and should record the number of trials before the user enters the correct guess.

Here is a sample run:

*The random guess is 50.*

|  |  |
| --- | --- |
| **Input** | **Output** |
| 90 | Too High (1) |
| 45 | Too Low (2) |
| 52 | Too High (3) |
| 50 | With 4 attempts, you got it ! |

**Problem 4**

Write a function that checks whether two words are anagrams.

Two words are anagrams if they contain the same letters. For example, **silent**

and **listen** are anagrams. The header of the function is:

**def** isAnagram(s1, s2):

(Hint: Obtain two lists for the two strings. Sort the lists and check if two lists

are identical.)

Write a test program that prompts the user to enter two strings and, if they

are anagrams, displays YES (All Capital) if **is an anagram**; otherwise, it displays NO (All Capital) if **is not an anagram**.

**Problem5**

Some Web sites impose certain rules for passwords. Write a function that checks whether a string is a valid password. Suppose the password

rules are as follows:

* A password must have at least eight characters.
* A password must consist of only letters and digits and no spaces.
* A password must contain at least two digits.

Write a program that prompts the user to enter a password and displays **valid**

**password** if the rules are followed or **invalid password** otherwise.

Input output

Mario1 invalid

Mariochahoud12 valid

Mario12 invalid

Mario 12 invalid

**Problem6**

**A good way to learn a language and how it works is to implement some of its built-in functionality. Do not use any of the List methods.**

* Write a function named **total(a\_list)** that behaves exactly like the built-in function sum().

>>> total( [1, 100, 3, 4, 5, 6] )

119

* Write a function named **contains(a\_list, item)** that behaves exactly like the built-in operator item **in** a\_list.

>>> contains( ['alice', 20, True], 20 )

True

>>> contains( ['alice', 20, True], 'eugene' )

False

* Write a function named **find(a\_list, item)** that behaves exactly like the built-in list method find().

>>> find( ['alice', 20, True], True )

2

>>> find( ['alice', 20, True], 42 )

-1

**Write a driver program that tests the above 3 functions. That is would call each function at least two times.**

**Recall:**

**Attention!**

**Policy on Cheating and Plagiarism:**

**Plagiarism on assignments and project work is a serious offense. If plagiarism is detected, a student will be subject to penalty, which ranges from receiving a zero on the assignment concerned to an “F” in the course.**