LAB: Artificial Intelligence

Lab Instructor	Ms. Saba Aslam
Department	Computer Science

Python Lab: Python Basics to Advance

Instructions

1. Make a word document paste your code and output there.
2. Comments in the code explaining chunks of the code is important.
3. Plagiarism is strictly prohibited; 0 marks would be given to students who cheat.
4. Late submissions are not allowed.
5. For the code part make sure your code is error-free and running. Failed to compile code will lead to zero marks in that question.
6. For coding part you can use Pycharm .
7. **For programming questions, you cannot use any built-in functions to perform the complete task .**
8. Increase the readability of your code by writing meaningful names of functions and variables.

Coding Part

1. Create a python program that takes your roll number and name. Then it greets you by displaying your name and roll number.
2. Take two numbers from the user as input and display the product of each.
3. Create a Calculator program with functions for addition, multiplication, subtraction and division of two numbers. It should handle the exceptions.
4. Create python program to take an input string from the user and display the words in the string separately using split () method each on new line.
5. Implement a program that calculates the sum of the cubes of the first n odd numbers entered by the user. The program should use a user-defined function and a for loop.
6. Write a program that takes a string as input and returns a string that is the reverse of the input string.
7. Write a Python program that accepts the user's first and last name as input and prints them in reverse order with a space between them. Additionally, the program should check if the input contains any numbers or special characters, and if so, return an error message saying "Input contains invalid characters." The program should also check if the input is empty or only contains spaces, and if so, return an error message saying "Input cannot be empty or only contain spaces."

   Example:

   Enter your first name: John

   Enter your last name: Doe

   Output: eoD nhoJ

8. Write a Python program to extract all the odd numbers from a list of numbers and return a new list of tuples, where each tuple contains the odd number and its index in the original list. Use a list comprehension to accomplish this

   Example:

   Input: [1, 2, 3, 4, 5, 6, 7, 8, 9]

   Output: [(1, 0), (3, 2), (5, 4), (7, 6), (9, 8)]

9. Write a Python program to display the examination schedule. The program should accept the examination start date as a tuple in the format (dd, mm, yyyy), and return the date in the following format: "The examination will start from : dd/mm/yyyy". Additionally, the program should validate the input to ensure that the day, month, and year are within the correct range (e.g. day is between 1 and 31, month is between 1 and 12, and year is between 1000 and 9999). If any of

the input values are out of range, the program should return an error message saying "Invalid input."
Example:
Input: (11, 12, 2014)
Output: The examination will start from: 11/12/2014.

Input: (40, 12, 2014)
Output: Invalid input.

10. Write a Python function to find the maximum and minimum numbers from a sequence of numbers. Additionally, the function should handle an input list that could contain both positive and negative numbers, and handle the case where the list is empty.
Example:
Input: [3, 4, -1, 5, 2, 6, 9, -3]
Output: Maximum: 9, Minimum: -3

Input: []
Output: Maximum: None, Minimum: None

11. Write a Python Program to convert a 2D array into 1D array. Your program should use the following functions.

i. A function to input the two-dimensional array.
ii. A function to convert the two-dimensional array in a one-dimensional array
iii. A function to print the converted array.

12. Given an unsorted array whose elements are all 0 (zeros) or 1 (ones), write code to sort the array so that all the 0's appear first, followed by all the 1's.

13. Given: A string of length at most 10000 letters.
Return: How many times any word occurred in string. Each letter case (upper or lower) in word
Sample Dataset:
We tried list and we tried dicts also we tried Zen
Sample Output :
and 1
We 1
tried 3
dicts 1
list 1

we 2
also 1
Zen 1

14. A string is simply an ordered collection of symbols selected from some alphabet and formed into a word; the length of a string is the number of symbols that it contains. An example of a length 21 DNA string (whose alphabet contains the symbols 'A', 'C', 'G', and 'T') is " AGCTTTTCATTCTGACTGCAA"
Given: A DNA string s of length at most 1000 nt.
Return: Four integers (separated by spaces) counting the respective number of times that the symbols 'A', 'C', 'G', and 'T' occur in s.
Sample Dataset
AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAA
AAGAGTGTCTGATAGCAGC
Sample Output

```
20 12 17 21
```

15. Write a function that takes a string as input and returns a string with all the punctuation removed.
16. Write a program that uses the datetime module to print the current date and time.
17. Write a program to randomly generate a list with 5 numbers, which are divisible by 5 and 7 , between 1 and 1000 inclusive.
18. With two given lists [1,3,6,78,35,55] and [12,24,35,24,88,120,155], write a program to make a list whose elements are intersection of the above given lists.
19. Write a program which prints all permutations of [1,2,3].
20. Implement a class hierarchy to represent a library system. The base class `Book` should have the following attributes:

- title (string)
- author (string)
- ISBN (string)

The class should have the following methods:

- check_availibility(): Return a string indicating whether the book is available for checkout.

Create a subclass `Textbook` that inherits from `Book`. `Textbook` should have the following additional attribute:

- edition (int)

Create another subclass `Novel` that also inherits from `Book`. `Novel` should have the following additional attribute:

- genre (string)

Implement a class `Library` that contains a collection of books. The class should have the following methods:

- add_book(book): Add a book to the library collection.
- check_out(ISBN): Check out a book from the library based on ISBN. If the book is not available, raise a "ValueError" with the message "Book not available".
- check_in(ISBN): Check in a book to the library based on ISBN. If the book is already checked in, raise a "ValueError" with the message "Book already checked in".
Also implement a menu-based driver function for the system to add remove or display books and their status in the library.