

Department of Information Technology – University of the Punjab
Programming for AI – MPhil/PhD (AI) F22
Lab-05

Max Time: 2.5 hours

Date: 18-1-2023

Topics: Problem solving, operators, basic constructs, selection, repetition

Instructions:

- Please provide your own solutions and **DO NOT COPY** the code from your colleagues or the web.
- You can discuss your problems only with the teachers.
- All tasks carry equal points.

Task 1

[20]

(Inheritance and Polymorphism)

Write a Python class named "Shape" that keeps data attributes for the following pieces of information:

- Shape area

Then create a "Circle" class that inherits from the Shape class and has an additional attribute "radius". Implement a method "getArea" in the Circle class that calculates and returns the area of the circle using the formula $\pi * \text{radius}^2$.

Next, create a "Rectangle" class that also inherits from the Shape class and has additional attributes "length" and "width". Implement a method "getArea" in the Rectangle class that calculates and returns the area of the rectangle using the formula $\text{length} * \text{width}$.

Next, create a function "printArea()" outside the classes. The function should take a shape object of any kind and prints the area.

Once you have written the classes and the print function, write a main program/function that creates objects of the Circle and Rectangle classes, prompts the user to enter the radius and/or length and width for each object, and then uses the "printArea" to display the area of each shape on the screen.

Task 2

[30]

(Sliding Window)

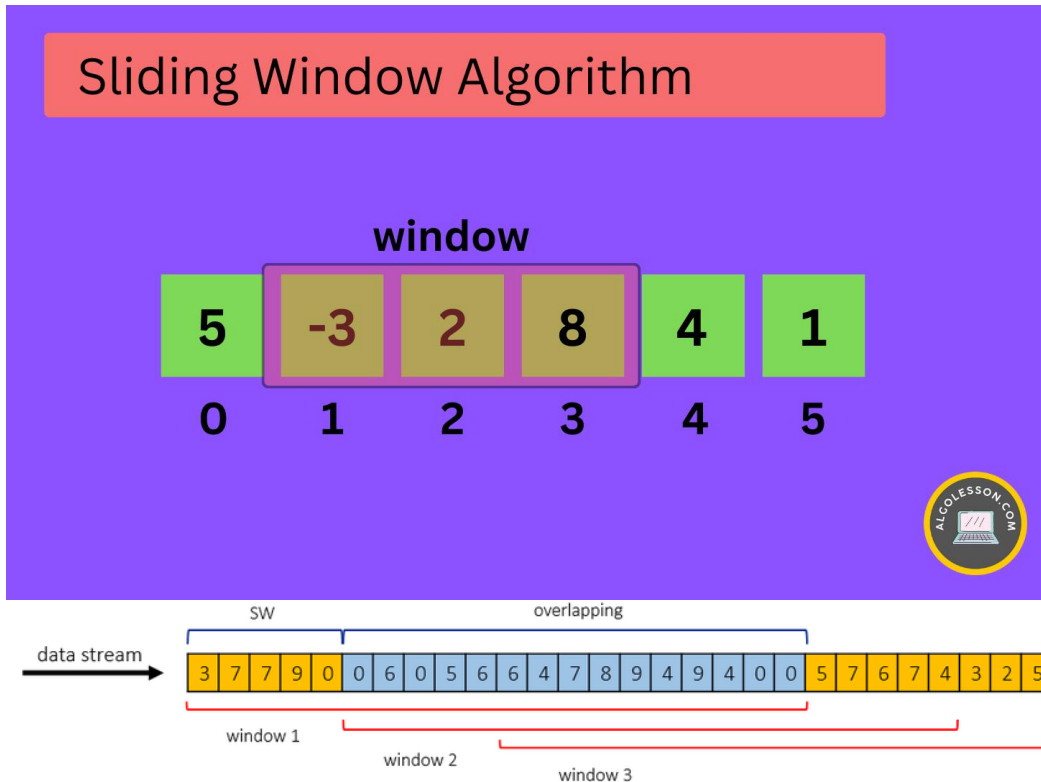
Given a 3D array (lab5_task2_arr.npy), suppose you have a sliding window of size $K * K * K$, which you are sliding over the array starting from index 0. The step size s of the window is also given. Step size refers to the number of index windows that move forward when slided once. You are required to slide the window over the given array and print the data in the window for each step.

For example, for window size $K = 2$ and step size $= 1$, the data of the window in step 1 will be

```
[[ [-102.5          -1.12804878]
   [ -69.9085388    -3.87195134]]
 [[  8.81097603     14.4817076 ]
   [  8.8719511     14.2682934 ]]]
```

While, in Second step it will be

```
[[[-1.12804878 -45.8841476 ]  
 [-3.87195134 -48.0182953 ]  
 [ 14.4817076  -14.3902445 ]  
 [ 14.2682934  -14.5121956 ]]]
```



Task 3

[30]

3D Array

You are given a 3D array arr. Create the following function.

1. Function that takes a 3D array and a number as input and returns a new 3D array where all elements that are greater than the input number are replaced with 0.
2. Function that takes a 3D array and returns a new 3D array where each element is the product of the corresponding element in the input array and its position (i.e. element at position (i,j,k) in the new array is equal to element at position (i,j,k) in the input array multiplied by ijk).
3. Function that takes a 3D array and two numbers as input and returns a new 3D array containing all elements of the input array that are in the columns between the two input numbers (inclusive).

Note: The following task is mandatory for PhD students. However, MPhil students can perform this task as a bonus task.

Task 4

[30]

Sudoku Verification

Sudoku is an interesting number-placement puzzle and challenging programming problem. You are given a solution to a simplified version of it and your task is to verify if it is correct or not. It is a 9 * 9 grid divided into smaller 3 * 3 boxes. A valid solution has unique numbers(between 1-9 inclusive) in every row, column and box. A valid sudoku solution is

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

An invalid sudoku solution is

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