LAB TEST

Instructions

- Make sure that your code is error-free and final script files should run without any warning/ errors.

- The order of execution/ output of the program should be same as mentioned in the provided use cases.

- Submit the complete code in a zipped folder named **<your ID>_Q2.zip**.

Question 2   [10 M]

You are provided a partial code with two files including **"pattern.h"**, and **"Q2.c"**. The file **"pattern.h"** contains global variables and function declarations that should be defined in **"Q2.c"**. The file **"Q2.c"** is incomplete with a local 2D array **arr** and sequenced function calls. The functioning of the code should first consider printing **arr**, followed by printing only selected elements in a half pyramid pattern. For example, if the number of rows is five, the code should first print the entire array **arr** -

```
1    2    3    4    5
6    7    8    9    10
11   12   13   14   15
16   17   18   19   20
21   22   23   24   25
```

followed by a half pyramid pattern where specific elements of arr needs to be considered. The half pyramid pattern should look like -

```
1
6    7
11   12   13
16   17   18   19
21   22   23   24   25
```

You are required to submit a zipped folder (named **<your ID>_Q2.zip**) with error-free code covering implementation of all functions in **Q2.c** as per the specifications given in the header file **"pattern.h"**. Kindly note the below points:

- The header file **"pattern.h"** is complete and you are not supposed to make any modifications to it.

- The functions **"print_array"** and **"print_pattern"** receives a pointer to the 2D array. Kindly note that he implementation of the code should be accomplished using pointer arithmetic only. Using direct array indexing is strictly not recommended and will not be considered for marking.

Example use-cases are provided below.

Use-case I	Use-case II
Enter the number of rows: 3 **Complete array** 	1	2	3
4	5	6
7	8	9	 **Array with pattern** 	1
4	5
7	8	9		**Enter the number of rows: 5** **Complete array** | 1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25	 **Array with pattern** 	1
6	7
11	12	13
16	17	18	19
21	22	23	24	25	