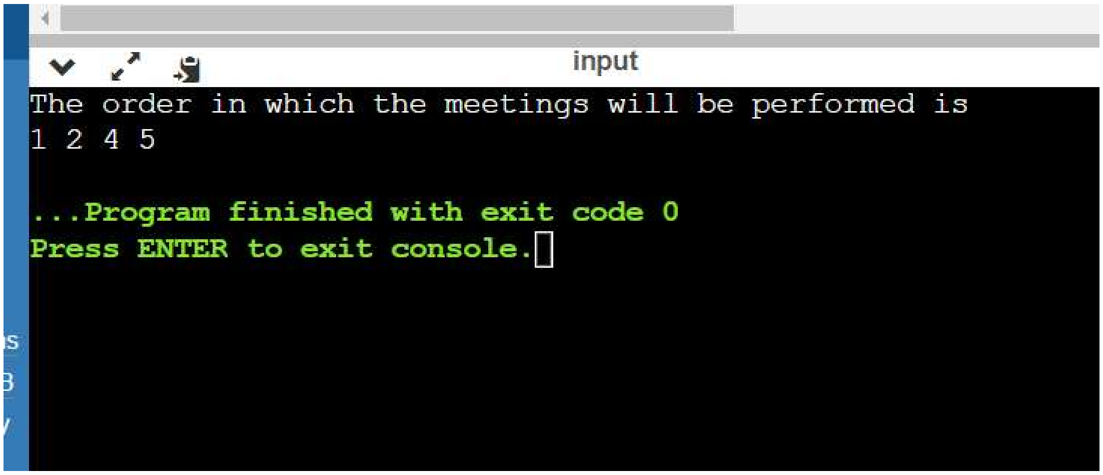
Day – 8 Greedy Algorithm

Problem 1: There is **one** meeting room in a firm. You are given two arrays, start and end each of size N.For an index ‘i’, start[i] denotes the starting time of the ith meeting while end[i] will denote the ending time of the ith meeting. Find the maximum number of meetings that can be accommodated if only one meeting can happen in the room at a particular time. Print the order inwhich these meetings will be performed.

```
from typing import List


class meeting:

    def___init__(self, start, end, pos):

        self.start = start

        self.end = end

        self.pos = pos



def maxMeetings(s, e, n) :

    meet = [meeting(s[i], e[i], i + 1) for i in range(n)]

    sorted(meet, key=lambda x: (x.end, x.pos))

    answer = []

    limit = meet[0].end

    answer.append(meet[0].pos)

    for i in range(1, n):

        if meet[i].start > limit:

            limit = meet[i].end
            answer.append(meet[i].pos)
    print("The order in which the meetings will be performed is ")
    for i in answer:
        print(i, end=" ")




n = 6
start = [1, 3, 0, 5, 8, 5]
end = [2, 4, 5, 7, 9, 9]
maxMeetings(start, end, n)
```



```
The order in which the meetings will be performed is
1 2 4 5

...Program finished with exit code 0
Press ENTER to exit console.
```