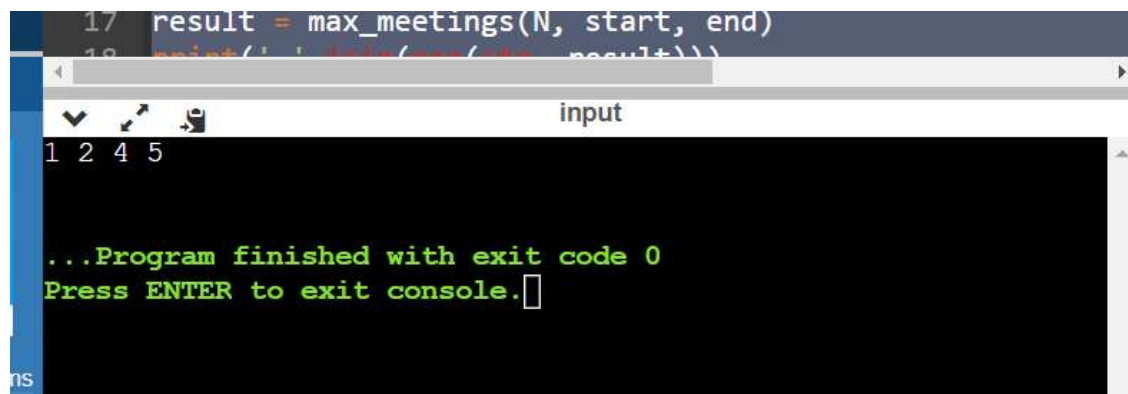


**Problem 6:** There is **one** meeting room in a firm. You are given two arrays, start and end each of size N. For an index 'i', start[i] denotes the starting time of the ith meeting while end[i] will denote the ending time of the ith meeting. Find the maximum number of meetings that can be accommodated if only one meeting can happen in the room at a particular time. Print the order in which these meetings will be performed.

```
def max_meetings(N, start, end):  
    meetings = [(start[i], end[i], i+1) for i in range(N)]  
    meetings.sort(key=lambda x: (x[1], x[0]))  
  
    selected_meetings = []  
    previous_end = 0  
    for meeting in meetings:  
        start_time, end_time, index = meeting  
        if start_time > previous_end:  
            selected_meetings.append(index)  
            previous_end = end_time  
  
    return selected_meetings  
  
N = 6  
start = [1, 3, 0, 5, 8, 5]  
end = [2, 4, 5, 7, 9, 9]  
result = max_meetings(N, start, end)  
print(' '.join(map(str, result)))
```



```
17 result = max_meetings(N, start, end)  
18 print(' '.join(map(str, result)))  
  
input  
1 2 4 5  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```