SRM INSTITUTE OF SCIENCE & TECHNOLOGY

DEPARTMENT OF NETWORKING & COMMUNICATIONS

18CSC305J-ARTIFICIAL INTELLIGENCE

SEMESTER – 6

BATCH-2

REGISTRATION NUMBER	**RA1911003011019**
NAME	**RAVI MYTRESH**

B.Tech- CSE / CC, Third Year (Section: H2)

Faculty In charge: Dr. S. Prabakeran, B.Tech, M.E, PhD
Assistant Professor
School of Computing - Department of
Networking and Communications

Year 2021-2022 / Even Semester

INDEX

Exercise: 1

Date : 4-01-2022

TOY PROBLEM

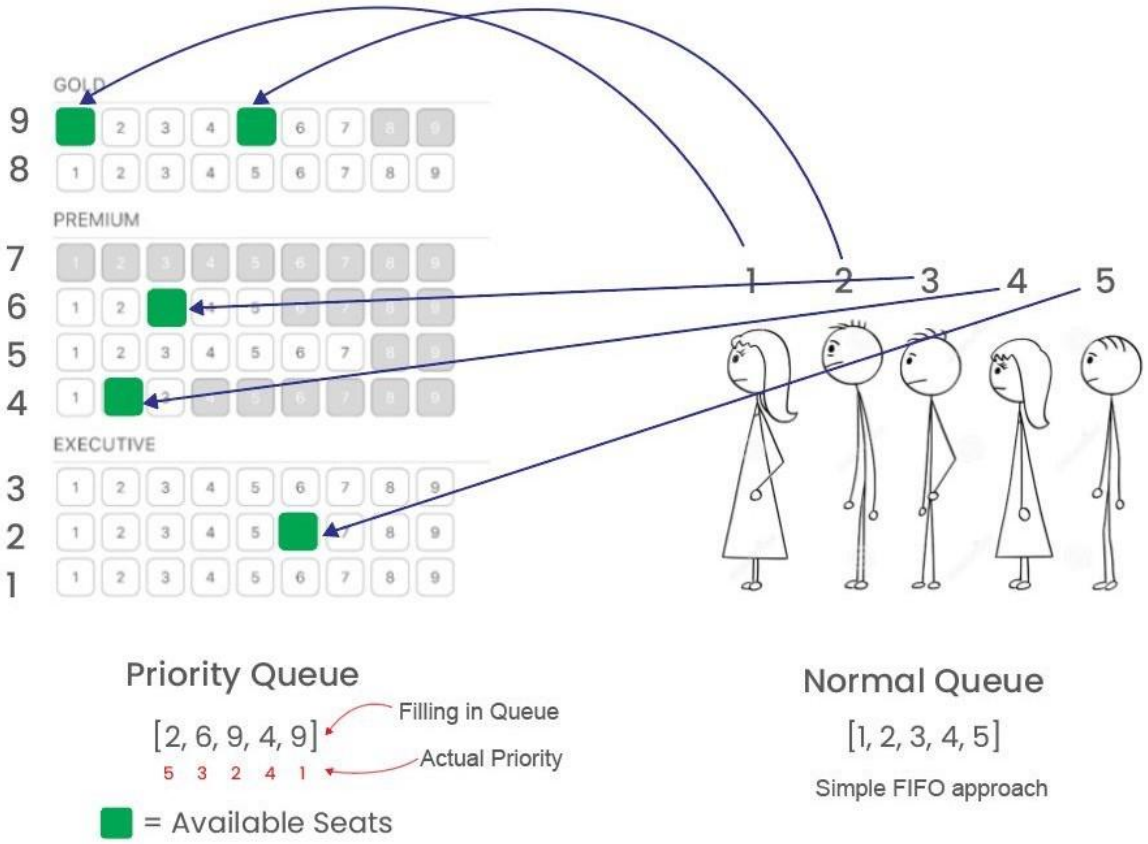
Problem Statement : Given an integer N and an array of seats[] where N is the number of people standing in a line to buy a movie ticket and seat[i] is the number of empty seats in the ith row of the movie theater. The task is to find the maximum amount a theater owner can make by selling movie tickets to N people. Price of a ticket is equal to the maximum number of empty seats among all the rows.

Algorithm :

1. Initialize queue q insert all seats array elements to the queue.
2. Tickets sold and the amount generated to be set to 0.
3. If tickets sold < N (People in the queue) and q top > 0
4. Then remove top element from queue and update total amount
5. Repeat step 3 and 4 until tickets sold = number of people in the queue.

Optimization technique : This problem can be solved by using a priority queue that will store the count of empty seats for every row and the maximum among them will be available at the top.

1. Create an empty priority_queue q and traverse the seats[] array and insert all elements into the priority_queue.
2. Initialize two integer variable ticketSold = 0 and ans = 0 that will store the number of tickets sold and the total collection of the amount so far.
3. Now check while ticketSold < N and q.top() > 0 then remove the top element from the priority_queue and update ans by adding top element of the priority queue. Also store this top value in a variable temp and insert temp – 1 back to the priority_queue.
4. Repeat these steps until all the people have been sold the tickets and print the final result.



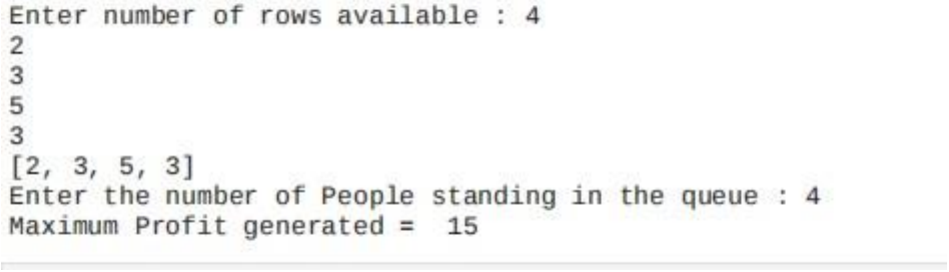
Tool : jupyter notebook

Programming code :

```
def maxAmount(M, N, seats):
    q = []      for i in
range(M):
        q.append(seats[i])
ticketSold = 0      ans =
0
    q.sort(reverse = True)      while
(ticketSold < N and q[0] > 0):
        ans = ans + q[0]
temp = q[0]         q =
q[1:]
        q.append(temp - 1)
        q.sort(reverse = True)
ticketSold += 1     return
ans

if __name__ == '__main__':
    seats = []     rows = int(input("Enter number of rows
available : "))     for i in range(0, rows):         empty =
int(input())         seats.append(empty)
print(seats) M
= len(seats)
N = int(input("Enter the number of People standing in the queue : ")) print("Maximum
Profit generated = ", maxAmount(N, M, seats))
```

Output screen shots :



```
Enter number of rows available : 4
2
3
5
3
[2, 3, 5, 3]
Enter the number of People standing in the queue : 4
Maximum Profit generated =  15
```

Result : Successfully found out the maximum amount the theater owner can make by selling movie tickets to N people for a movie.