
SRM INSTITUTE OF SCIENCE & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**18CSC302J-COMPUTER NETWORKS**

SEMESTER – 6

BATCH-2

| | |
|---|---|
| **REGISTRATION NUMBER** | **RA1811028010049** |
| **NAME** | **SHUSHRUT KUMAR** |

**B.Tech-CSE-CC, Third Year (Section: J2)**

**Faculty Incharge: Vaishnavi Moorthy, Asst Prof/Dept of CSE**

**Year 2020-2021**

# INDEX

**Exercise: 1**

**Date : 21-01-2020**

# 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 :** VS Code and Python 3.9.0

**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 :**


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