

MOVIE THEATER SEATING CHALLENGE

Coding language:: C++

Problem Statement:

Implement an algorithm for assigning seats within a movie theater to fulfill reservation requests. Assume the movie theater has a seating arrangement of 10 rows x 20 seats. The algorithm should maximize customer satisfaction and customer safety. For customer safety, a buffer of three seats is required.

Layout movie theater

```

                SCREEN
            -----
            1.....20
A
B
C
D
E
F
G
H
I
J
```

Algorithm:

1. The movie theater I'm considering here is a cinema screening hall.
2. Customers with early reservations are given priority seating i.e, topmost rows (away from the screen) will be allocated.
3. All the customers under the same reservation ID are allocated seats together in the same row.
4. Each group will be allocated seats in the same row with a name reservation ID. If the number of customers in a group reservation is greater than the seats available then split the group and allocate the customers in the available rows.
5. If the number of seats requested is not available, insufficient seats in the theater message are informed to the customer.
6. For customer Safety, there is a buffer of three seats between different groups (the customers with different reservation IDs are considered as different groups).
7. Allocate all the possible vacant seats to the customer if consecutive seats in the same row are not available.

Goals of problem achieved.

- Customer Satisfaction is achieved by allocating consecutive seats to the customer from the same group and trying to accommodate as many groups as possible. In case seats in the same row are not possible provide seats anywhere in the hall.
- Customer Safety is achieved by providing three "Do Not Sit"(DNS) seats between groups.
- Maximum Utilization of the movie theater is achieved by accommodating as many groups as possible and satisfying them by providing consecutive seats starting from the first column. This will help in maximum occupancy in the theater.

Steps for running the program.

1. Download the code.
2. From the terminal take the path to the directory and build the code and then run the code
 - a. `./file_name "path_input file"`
3. The output file is saved in the same directory and the path to the output file is seen on the command line.