

1. Input the dimensions of the theatre:

- Prompt the user to enter the number of rows (`n`) and columns (`m`) for the theatre.

2. Initialise the theatre grid:

- Create a 2D array `arr` representing the theatre with all seats initially set to 0 (unoccupied).

3. Print the initial state of the theatre:

- Display the empty theatre grid.

4. Seat Assignment Loop:

- Begin a loop that continues until the user inputs a negative number.
- Prompt the user to input the number of people (`x`). If `x` is negative, exit the loop.

5. Seat Assignment Process:

- Initialize variables:
 - `seek_row` to track the row where seats are being allocated.
 - `last_zero_id` to store the index of the last unoccupied seat in the current row.
 - `recent_updates` array to track the latest seat assignments.

6. Iterate through the theatre rows:

- Start a loop to go through each row of the theatre.
- Search for the last unoccupied seat in the row.
- If the number of people is greater than the available seats in the row, move to the next row.
- If all rows are checked and there's no sufficient space, display a message and exit the program.

7. Assign seats:

- If enough empty seats are available in the row:
 - Mark the consecutive seats as occupied (`1`).
 - Update `recent_updates` array with the row and column indices of the assigned seats.
 - Print the updated theatre grid.
 - Display the assigned seats in a human-readable format (`A1`, `B3`, etc.).

8. Repeat

9. Exit the program:

- When a negative number is entered or no empty place is left, display an exit message and terminate the program.