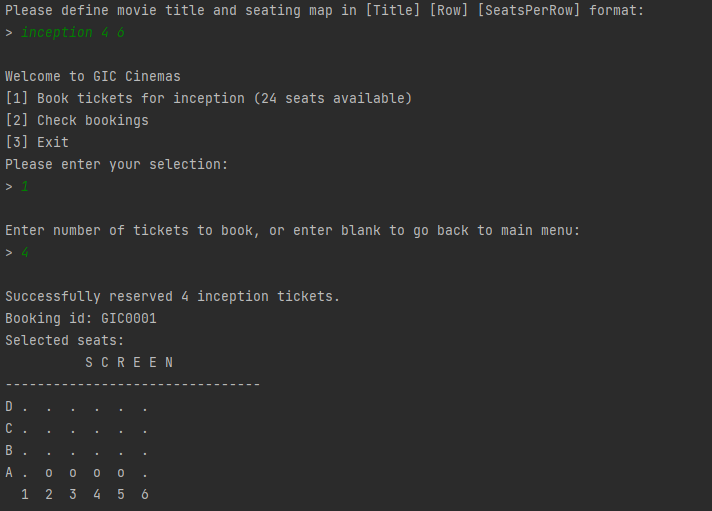
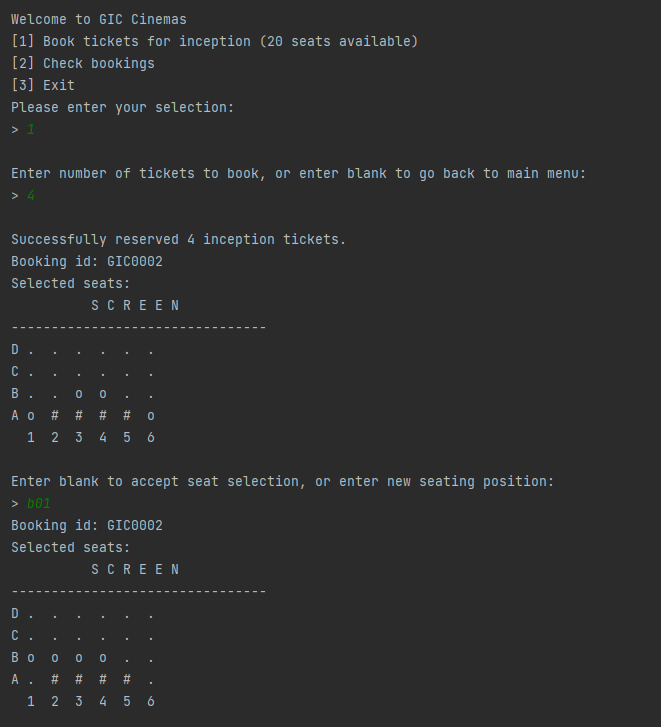
Summarizing the booking process:

Cinema class is designed to simulate seat bookings in a cinema with thread-safety mechanisms to handle concurrent seat reservations. It uses a global lock (self.lock) around the entire booking process to ensure that only one thread can modify the seating arrangement at a time. This lock prevents race conditions when multiple threads attempt to book seats simultaneously, ensuring that the available seat count remains consistent. The class offers methods for displaying the cinema layout, allocating seats based on different strategies (default or custom), and confirming bookings. Thread safety is achieved by locking the entire booking operation, but a more efficient approach could be to implement **seat-level granularity**, where each seat has its own lock. This would allow threads to book different seats concurrently without waiting for access to the entire cinema grid, improving performance and reducing contention when multiple threads are trying to book independent seats.

Booking follows the alorithm

1. Booking: Begin at the back row, farthest from the screen.
2. Start seating from the middle column(s) of that row.
3. Fill seats outward from the middle.
4. If a row is full, move to the next row closer to the screen.
5. Repeat steps 2-4 until all tickets are assigned.





Retrieval:

