

## Queue

**Time span:** 1.5 lab days(3 hrs)

**Problem:** Development of ticket booking module for a day of typical Cinema hall.

**Description:** A typical Nepali cinema has single hall and has two types of reservation for a particular show.

- Family Seat Reservations of size  $n_1$ .
- Single Seat Reservations of size  $n_2$ .

So, the Total seats available =  $n_1 + n_2$ .

The policy for ticket booking follows first come first served basis. i.e. first single person gets ticket no. 1 of single type, second gets ticket no. 2 of single type and so on. This process can go up-to  $n_1$  persons.

Similarly, first family gets ticket no. 1 of family type, second gets ticket no. 2 of family type and so on. This process can go up-to  $n_2$  families.

A person with a phone number can make the ticket reservation. The ticket booking system resets the ticket counter to begin from 1 in every show.

**Sample data input:**

Phone Number	Ticket Type	Arrival Time
+977-9849023236	Individual	10 Unit
+977-9849023260	Family	21 Unit
+977-9849023299	Family	22 Unit
+977-9849023220	Individual	24 Unit
+977-9849023243	Family	27 Unit

### Solution:

1. Design and implement two Queues that can handle enqueue, dequeue operations for single and family types of ticket reservations. You need to check a pre-condition if the given phone number has already booked a ticket or not.

2. Write a function that returns the aggregated number of available seats of individual and family types reservation.

Input:  
Output: totalAvailableSeats

3. Write a function that takes 'phoneNumber' as input and returns the ticket number.

Input: phoneNumber  
Output: ticketNumber

4. Write a function that takes 'phoneNumber' as input and returns the ticket type.

Input: phoneNumber  
Output: ticketType

5. Write a function that calculates the maximum time difference between consecutive ticket allocations.

### Assumptions:

Make your necessary assumptions for  $n_1$  and  $n_2$  with relevant data data types to represent phone number (char\* or your choice), ticket(int) and arrival time(long). A person with a single phone number can make a booking of a single ticket only for a show.

\*\*\*