```
/
*
    * Created by: Maxwell Meckling
    */



    #ifndef TRAINSEATING_H
    #define TRAINSEATING_H


    #include <string.h>
    #include <stdio.h>
    #include <stdlib.h>
    #include <sys/ipc.h>       // Used by IPC maechanisms: messages, shared memory
    and semaphores
    #include <sys/shm.h>  // Possibly also for shared memory
    #include <sys/mman.h>   // Needed for shared memory
    #include <sys/stat.h> //I don't think this one helps
    #include <fcntl.h>      // Needed for shared memory
    #include <semaphore.h>  // Used for semaphores
    #include <stdbool.h>    // Used to declare boolean values
    #include <unistd.h>     // Used for ftruncate in shared memory


    #include <sys/types.h> //need these for sockets
    #include <sys/socket.h>
    #include <arpa/inet.h>
    #include <netinet/in.h>


    #include "andrew_trainTicketMaster.h"
```

```
#define STRING_BUFFER_MAX 300//for tcp



//calculates and then returns the number of available seats for a given day
int countNumberOfAvailableSeats(availableSeats*);


//Returns 0 or 1 based on the passed dayOfTravel variable to match which day we
should be on.
//Looks at the int, not the string
int matchDayOfTravel(availableSeats*, int);


//helps with sending a message through tcp
void seatingSendMessageToClient(char*, int);


//accesses shared memory to assign next available ticket number to customer
//then increments ticket number for next customer
//returns int ticket number
int assignTicketNumber(customerInfo, int, availableSeats*);


//checks shared memory using customers numberOfTravelers
bool checkIfAvailableSeats(int, int, int, availableSeats*);


//shows seats customer selects starting index (seat) and #of travelers fills in
seats
//accessess shared memory to read seats available and copies to string buffer
and then sends to client via tcp
void displayAvailableSeats(int, int, int, availableSeats*);
```

```
//accesses shared memory and alows customer to select from available seats and
writes to shared memory and saves seats to customer struct copy
//will use int nextCustomer.dayOfTravel and mextCustomer.numberOfTravelers

//had to add addedSeatsIf Modified for when just adding select number number of
seats
customerInfo selectAvailableSeats(customerInfo, int, int, availableSeats*);


//Frees the customer's seats and frees those seats in shared memory too

customerInfo freeCustomersSeatsInSharedMem(customerInfo, int, int,
availableSeats*);


#endif /* TRAINSEATING_H */
```