

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <stdbool.h>
4  #include <string.h>
5  #include <ctype.h>
6  #include <pthread.h>
7  #include <sys/types.h>
8  #include <unistd.h>
9  #include <sys/wait.h>
10 #include <sys/socket.h>
11 #include <netinet/in.h>
12 #include <arpa/inet.h>
13 #include <semaphore.h>
14 #include <fcntl.h>
15 #include <sys/stat.h>
16 #include <errno.h>
17 #include <time.h>
18
19 #ifndef header
20 #define header
21
22 #define NUM_THREADS 3
23
24 struct clientInformation{
25     char ClientName[50];
26     char DateOfBirth[50];
27     char Gender[10];
28     int GovernmentID;
29     char DateOfTravel[50];
30     int NumberOfTravelers;
31     int MenuOption;
32     char seats[100];
33     int ticket;
34     char modified[200];
35     int server;
36 }__attribute__((packed))clientInformation;
37
38 struct customer_queue {
39     pthread_t threads[NUM_THREADS];
40     int seats_for_thread[2*NUM_THREADS];
41     int sockets[100];
42     int waiting;
43     int first;
44     int port;
45 };
46
47 // AdditionalFunction.c
48 int seatChecker(int trainNum);
49 int GetTomorrowDate(char *tomorrowsDate);
50 int GetTodayDate(char *todaysDate);
51
52 // server.c
53 int create_socket(int port, struct sockaddr_in* address);
54 int initialize_semaphores_threads(struct customer_queue* q, int reset);
55 int thread_loop(void* args);
56 int serve_customer(int socket, int t_id, int s_id, int* seats_for_thread);
57 int get_customer_info(int socket, struct clientInformation* c);
58 int get_customer_ticket(int socket, struct clientInformation* c);
59 int get_train(struct clientInformation* c);
60 int change_read_count(int offset);
61 int signal_write(int train);
62 int wait_write(int train);
63 int signal_read(int train);
64 int wait_read(int train);
65 int check_thread_permission(int id, int train, int seats, int* seats_for_thread);
66 int verify_enough_seats(int socket, int train, struct clientInformation* c);
67 int confirm_purchase(int socket, int train, struct clientInformation* c);
```

```
68 int confirm_cancel(int socket, struct clientInformation* c);
69 int confirm_modify(int socket, struct clientInformation* c);
70 void show_available(int trainNum, char* output);
71 void send_available_seats(int socket, int train, struct clientInformation* c);
72 int check_seat(int train, int row, int column);
73 int verify_selection(int socket, int train, struct clientInformation* c, char* m);
74 int write_seat(int train, int row, int column, int update);
75 int add_to_train(int train, struct clientInformation* c, char* m);
76 int remove_from_train(struct clientInformation* c, int train);
77 int server_loop(int server_fd, int port, struct sockaddr_in* address, struct customer_queue* q);
78 int main(int argc, char const *argv[]);
79
80 // client.c
81 int connect_to_server(char* ip_addr, int port);
82 int main(int argc, char const *argv[]);
83
84 // Summary.c
85 void addCustomer(struct clientInformation *customer, int new);
86 void changeOldCustomer(struct clientInformation *customer);
87 void deleteCustomer(struct clientInformation *customer);
88 void printCustomerInfo(struct clientInformation *customer, char *output);
89 int findCustomer(struct clientInformation *customer);
90 void createCustomer(struct clientInformation *customer);
91 int findCustomerTicket(int ticket);
92
93 #endif
```