

The aim of the program is to emulate the workings of travel agencies booking seats on a flight, using multithreading. It includes three threads: the main thread, TA1, TA2. A matrix of seats, an integer 'turn' and a Boolean 'Full' are initialized as global variables as they are to be accessed by all the threads. Using `srand ()` provides random integers to be utilized as seat numbers. The matrix is initialized, all variables assigned as 0. Now a loop is initiated which can only be broken by the plane being full i.e. `bool Full` being true. This condition is checked at the start of each loop by the main thread. Then the 2 agency threads are created and busy waiting is used to prevent a race condition. The global variable 'turn' when equals to 0, allows the first thread to access the shared matrix and vice versa. The thread generates random values for row and column within the range of the matrix to obtain seat numbers to be booked. If the given seat is available, it is booked and the thread exits the critical region. Otherwise it continues to find an available seat. The thread also checks if the plane is full and in such a case it exits the critical region without making a booking. Once out of the main loop (when `bool Full` is set as true by main thread), the matrix representing the bookings is printed out.