

# CmpE 322 – Operating Systems

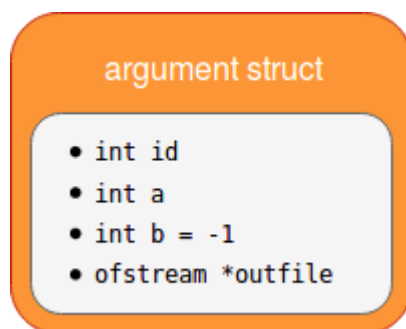
## Project#2 - Flight Reservation System Simulator

### Design and Implementation

I designed my flight reservation system simulator as follows:

- main thread creates client threads
- client threads sleep
- client threads creates their server threads

Client threads	Server threads
<ul style="list-style-type: none"><li>• client threads picks a seat (critical section)</li><li>• client thread invalidates the seat(critical section)</li></ul>	<ul style="list-style-type: none"><li>• server thread waits until a seat is selected by the client</li><li>• server thread reserves and logs the seat to the output.txt (critical section)</li></ul>



I used a struct to pass arguments to the threads and each client-server couple have their own argument struct. ID is not the actual id of the thread its just an integer variable to help me pair the server and client threads. If a server thread created by the client thread m, it has the id m as well. Integer a is the total number of seats, and int b is the seat selected by client thread, if none selected yet, its -1. The ofstream pointer is to log the output file easily.

### Compile and Run Commands for Linux

```
g++ main.cpp -o main -lpthread
```

```
./main <number of seats>
```