Ayça Avcı
53961

COMP 304
PROJECT #2
REPORT

I implemented all the parts in the given project pdf. All the parts work without any error.

I implemented the first part using the given pseudocode. First, I created a Plane struct which includes all the information about the plane such as status, ID, arrival time, lock, cond, permission etc. Since I implemented the project in C++, I used queue data structure that C++ supplies to store emergency, landing and departing planes. In the main function, I created threads for the air traffic control tower, departing planes and landing planes. At time 0, there are initially one landing and one departing plane. After every 1 second, a new plane is created according to a randomly generated probability value.

Every plane has its own lock, cond and permission values. All the planes are controlled by the ATC tower that decides which plane is going to the runway. When a plane arrives on one of the queues, it gets its own lock and waits for the permission from ATC tower. When permission is given to the plane, ATC sends a signal to the plane. It starts using the runway and its thread sleeps 2 seconds.

Every 40 seconds, if a landing plane is created, it becomes an emergency plane and is pushed to the emergency queue. Whenever the runway gets empty, it immediately uses the runway.

To solve the starvation problem, I put the below condition. If it is not an emergency plane:

- If landing queue is not empty and its size equals to or bigger than 10 and it is waiting more than s/8 seconds, or, departing queue is not empty and its size equals to or bigger than 5 and it is waiting more than s/10 seconds, function gets planes that are waiting front of the queues and compares their arrival time. The one whose arrival time is less than the other one will use the runway. With this implementation, no plane will wait more than s/8 or s/10 seconds depending on whether it is a landing or departing plane. The reason that I used different conditions for landing and departing planes is to create higher probability for this condition to hold. After the condition holds, it is 50% probability of which plane will use the runway. There is no longer any priority for any plane.
- If the above condition does not hold, priority will be in the landing plane, then the departing plane. Because of this, at the beginning of the simulation, ATC will favor landing planes for a while.

I printed the plane's ID, status, request time, runway time and turnaround time to the plane.log file. Also, I printed the waiting planes in the queue according to their IDs. It starts to print every second which planes are in the queues starting from time n. (n is initially set to 20 seconds). They are printed to the console.