

COMP304 - Operating Systems - README Report - Hakan Şahin (64355)

The project is about simulating an ATC tower at an airport with a single runway, where a plane will land or depart in every t seconds and an emergency plane has to land every $40t$ seconds. The main part of this project is that only one plane can use the runway at a time, so it is managing the runway usage without causing starvation in both departing and landing planes. There are two sample logs file with $p = 0.5$ $n = 20$ and $s = 60$ included.

How to Make the Code Run:

Compile the code with:

```
g++ -pthread -o air_traffic_control air_traffic_control.cpp
```

Run the code with:

```
./air_traffic_control -p 0.5 -n 20 -s 60
```

where p is the probability, n is the second after we start printing to the terminal, and s is the total simulation duration. You can also feed a seed to random number generator by:

```
./air_traffic_control -p 0.5 -n 20 -s 60 -seed [insert seed integer here]
```

Part I (Fully Working):

This part implements the main logic of the project. Where planes have chance to land/depart every t seconds. No emergency and starvation is yet implemented in this part.

Part II (Fully Working):

This part tried to solve the starvation that happens in the departing queue, since the first part causes departing planes to stack up, since the landing queue had priority for the ATC. I tried to solve this issue by making the final difference between two queues 1-4 planes, landing queue being less since it should still have priority due to fuel efficiency. I said in order for a plane to take depart, landing queue should be less than 12, departing queue size should be bigger than 0, departing queue's first plane's wait time should be bigger than $2t$, which resulted in a good results in $p=0.5$ and $t=60$.

Part III (Fully Working):

Implementation of this part was fairly easy as we only had to add emergency planes landing every $40t$ seconds. We only had to give priority to the emergency planes in the atc thread, so it can land first as soon as it arrives.

Keeping Logs (Fully Working):

I have implemented the logs part according to what was given in the PDF and what was provided in the Operating Systems Forum. It isn't pretty since it doesn't have GUI, but it gets the job done.

Things I Want to Add:

The implementation is thoroughly commented in the code, if some part is not made up clear here, there is a lot of comments found in the [air_traffic_control.cpp](#)