DUGNY Gabriel
May 2018

Efrei Paris
L1 Inter

Project 4

Airport Traffic Simulator

Introduction

The last project of the year was the one we found the most difficult. Indeed, it made use of all we've seen this year in class and lab. In this report, we'll see what the aim of the project was, the different difficulties we encountered and what we learned from this project.

Description of the project

The objective of this project Is easy to catch with its name: Airport Traffic Control. It is a simulation of an airport traffic control, where the user can interact with planes taking off, landing, with the different companies, etc.… The simulation takes care of landing the planes and making them takeoff on time, with only a single runway, depending on user events such as blacklisting a company or making a plane high-priority, but also random events, generated by the program.

Encountered difficulties

We have encountered a lot of difficulties to program this project. The biggest difficulty was understanding the structure of the program and implementing it. We had to understand the global structure of it, as well as all the structures used to implement the different components of the simulator, like planes or companies. This task was really time consuming because we had to go back to structures we've studied like linked lists, double linked lists or queues and modify them to suit our needs in this project. Memory management is something that created a lot of bugs and crashes at the beginning, and even if it works now, it still needs works to properly free the memory.

Another problem we haven't thought of was the user input. Indeed, C being low-level, even understanding takeoff and landing codes from the user was tricky. We figured that out eventually, by cleaning the input string for instance, but user input also needs work. We had a bug, where the `scanf` would skip half the time. With fixes, we now must press enter twice to enter a command, which is not what we wanted. We've implemented few checks on the user input, and it is possible to crash the program by typing an incorrect command.

These problems created a bigger one: we haven't got to the best part. Indeed, implementing the basics was way longer than expected, and we have not been able to implement some more advanced and interesting features, with the algorithms checking the priority for landing, takeoff, etc.…

Conclusion

We underestimated the project, because of a lot of smalls difficulties low-level programming languages can present. Nevertheless, this project was a great way to improve our C skills, using concepts we already knew in a bigger project. We hope to be able to achieve more in other projects, and not just a basic working prototype.