shreyOS

Introduction

To hone my C++ skills and to learn more about how operating systems work behind the scenes, I took up a project to write my own operating system, shreyOS. Not just did the project help me understand operating systems, it also helped me gain a deeper understanding of designing fairly complex projects in an object oriented way.

Tools

One of the most important aspects of starting a project is to choose the right tools and languages. In my case, although C is a more preferred language when writing operating systems, I chose to go ahead with C++ since handling such a complex project in C without an object oriented design pattern to the project was going to be tough. I also needed a linux distribution to work with, and I chose cinnamon mint, since I was already comfortable with it and had been working with it even before I thought of this project. I used VirtualBox for running the operating system, since it was an impractical idea to install the kernel and reboot the system to check out every change I made.

Once I had decided and set up my tools, I jumped into writing the actual OS.

Initial Work

My work started off with writing a simple kernel, the assembly code and a makefile for my project. Soon after, I added global and interrupt descriptor tables, and was able to handle various kinds of interrupts, including keyboard and mouse interrupts. I wrote drivers for both of these and was able to execute code based on the interrupts. At this stage I created two branches of the project. The master branch progressed to have a VGA interface, whereas the other branch was more focussed on running the operating system from the command line itself.

Graphics

I wrote a peripheral component interconnect to communicate with the different cards in the operating system. The most exciting one among these was the graphics card, and I jumped right into creating VGA graphics. Instead of a black screen with white text, I was now able to create a blue screen, and after I created a class for windows, I was able to get windows (my windows at this point were just rectangles of different dimensions). I was also able to show a mouse cursor, and my windows could be dragged around the screen using the mouse. This is where the graphics part of shreyOS stands right now, since making a

decent user interface would require me to write a complete GUI framework, which is a rather daunting task.

Commands

I modified my black screen to function more like a console, so that it showed a prompt and was able to recognize some basic commands. These commands are ‘echo’, ‘prompt’ and ‘clear’. The lack of a dynamic memory management system and a file system limits the number and type of commands that I can write, so the next step here is to add these. I also plan on adding drivers for the network card, so that the operating system can receive and transmit data over the network. This feature will not only improve my operating system but also give me a deeper understanding of networking and the interaction that happens between networks and processors.

Takeaways

Creating an operating system from scratch was an amazing learning experience and will continue to be whenever I work on improving it. My work on shreyOS taught me a lot about how operating systems work and how one should go about structuring a complex project in an object oriented manner. It also taught me the real importance of makefiles and how to write good, readable and maintainable ones. I could not anticipate how complex the project was going to become when I started off, and made the mistake of not using namespaces from the very start. If I were to work on a project again, using namespaces is one thing I would definitely take care of. The project also taught me a lot about masks for bitwise operations and the importance of operators such as the left and right shift, which are not normally used in simpler projects. I hope to keep working on shreyOS whenever I can devote time for it, and I’m sure I’ll be able to turn it into a fabulous operating system one day!