Design Project 3 Proposal

Yinhao Qian

Project Overview

I would like to build my own operating system kernel from scratch. This kernel can be directly booted from any x86 architecture. In this kernel, I'll attempt to reimplement some core features of operating systems such as interrupter, heap, paging, file systems, multi-threading manager, and etc. However, those are subject to change.

Since this is a pure software project, there should not be any schematics or drawing. Pseudocode is currently unavailable since this project will include thousands of lines of code once finished.

Required Resources

Since this is a software-based kernel, no additional resources are required. I would able to directly test the project on my PC.

My estimated budge would be 0 dollars. The only required resource needed is a functional GCC compiler.

Project Justification

Aside from data structures and algorithms, operating system basics are important for cracking the interview towards FAANG tech companies. Reimplementing the system kernel could reinforce my understanding towards operating systems.

I would include as many functionalities in my system core as possible if time allows.

Schedule

Here is going to be a preliminary schedule per half week:

- Bootloader Implementation
- Interrupts Implementation
- In and Out Function Implementation
- Heap Implementation
- Paging Implementation
- Disk Driver + File Manager Implementation
- Task Switching Implementation
- Executable and Linkable Format Implementation
- Standard C Library Implementation

The final video demonstration will show a fully functional operating system in a virtual machine.

Potential Roadblocks

I might reduce the number of implementations from schedule if I don't have enough time, but in that terms the system should still be functional but just without certain features. If one section has bugs or faults, I'll just disable that section.