

## Project 1

### Project Purpose:

This project's main goal was to build a simpler computer system simulation with an emphasis on the communication between memory management and CPU processes. The goal of the simulation was to show off how a CPU works, including how it handles interrupts, executes instructions, and communicates with memory. In order to safeguard system memory, it also attempted to demonstrate the fundamentals of memory management, such as reading from and writing to memory addresses and separating user and kernel modes.

### Project Implementation:

The project was implemented in C++, utilizing Unix system calls like `fork()` for process creation and `pipe()` for inter-process communication. The program starts by creating two pipes for bi-directional communication between the CPU and memory processes. The CPU process simulates the execution of various instructions, such as loading values into registers, performing arithmetic operations, and handling system calls and interrupts. It communicates with the memory process to read from and write to memory addresses. The memory process, initialized with data from a file, responds to requests from the CPU process by providing the required data or performing the requested memory operations. The implementation includes mechanisms for handling memory violations, simulating timer interrupts for preemptive multitasking, and exiting the simulation gracefully upon executing a specific instruction.

### Personal Experience:

Working on this project gave me a deep understanding of computer systems' internal workings, especially the intricacy of memory management and CPU operations. A thorough understanding of how a CPU understands and executes instructions, maintains registers, and communicates with memory was necessary to implement the simulation from scratch. It also required an understanding of creating and communicating processes in an environment similar to Unix, which was difficult but incredibly gratifying. The research demonstrated the value of thorough design and debugging techniques since accurate and dependable software simulation of hardware behavior necessitated close attention to detail. All things considered, the project was a worthwhile educational opportunity that improved my comprehension of system programming and computer architecture.