**Rationale**

In developing this mock shell in C, my goal was to understand the fundamental concepts of Operating Systems, with a focus on process management, and inter-process communication or IPC. Using functions such as fork(), and handle\_pipe() were intentional and they demonstrate the essential aspects of how an OS creates and manages processes. Understanding process management is crucial when we are speaking in the context of Operating Systems.

I created handle\_pipe to demonstrate basic IPC. By using the pipe() and dup2() system calls, I demonstrated how processes within an Operating System can communicate and synchronize data. This becomes vital when implementing multitasking operations which is a necessity in modern Operating Systems. I also implemented input/output redirection with the redirect\_input\_output() function. This function aims to mimic a critical functionality of shells, whereby it allows for the redirection of standard input and output streams to be read to and from a file. This feature shows how Operating Systems handle data between processes and files.

Command interpretation, showcased in functions like read\_command and type\_prompt, is key to understanding how shells process user inputs. This mirrors the role of an OS in providing an interface for user interaction, enabling the execution of commands and the management of responses. On the other hand, I demonstrate job control through the is\_background\_job() function. My intention here was to allow the shell to continue running commands but still have the ability to execute processes in the background. Lastly, I implemented error handling and process termination, by using perror and exit. These were incorporated to demonstrate best practices in robust programming.

Overall, I intended to create a simplistic mock shell that demonstrates the fundamental concepts of Operating Systems without delving too far into the weeds of complexity. I hope my code structure is clear and easy to understand as I put my best effort into making it as minimalist, but functional as possible.

Link to Loom Recording: