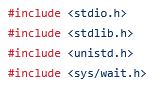
Lab 2 Report

Sean Gallaway

# Introduction:

Lab 2 is a lab about creating different processes and having them run independent tasks. The program is to be ran on a Linux machine and should be compiled using a Makefile. The main parent thread should wait to continue execution until after the child processes are finished with their tasks, in which the parent program will exit gracefully.

## Code explanation



These includes are essential to the program. Stdio.h is for reading and writing to the standard output, stdlib.h gives some general purpose functions, unistd.h allows us to use system calls like fork() and exec(), and sys/wait.h allows us to have the parent process wait for the child processes.

A screenshot of a computer code

AI-generated content may be incorrect.

Inside of the main function, variables are initialized such that the program can function. A list of command is created that will later be passed to the child processes to run with exevp.

A computer code with text

AI-generated content may be incorrect.

The following code block is the bulk of the code for the lab, the main process continually spawns child processes in a for-loop, if there is no error. If the fork was successful, the child runs its designated command, checks for error, then exits.

A screen shot of a computer

AI-generated content may be incorrect.

The parent process, after spawning the child processes, waits for the children to finish using the wait() command.

## Conclusion

This was an explanation for lab 2, which was important for me to be able to better understand processes.