Project 1

Changoh Hong

Dr. John Svadlenka

Throughout the coding, project1, I could learn analyze the process with system calls in the program. I learned not only one process can run but also multiple process can run at the same time and they can share the same terminal to print. I learned parent process can create the child process and child process also can create another children processes. This could be very useful because this allows multiple tasks to run independently of one another as though they each have the full memory of the machine.

For using fork() function, Child process is not brand new process that is created. It is just cloned from the parent process and two different process continue the execution and the child process’ PID is returned to the parent’s thread of execution, and a 0 is returned in the child’s thread of execution. However, if it fails to create the child process, -1 will be returned to the parent’s context.

I adapted the function from the previous project0 that can be called in the main function. And it was only utilized for parent process. In the main function, unlike the previous project0, 4 arguments are required because I need to make two target files, one for parent and another one for the child process. After the fork(), I set up 3 different situations, first, if pid<0, fork() fails, second, if pid is equal to 0, child is created, and third, if pid is greater than 0, the parent process and the child process is executed but only the parent process will execute. And if pid equal to 0 and child is created, it executes the cpfile.exe one that is from the previous project0 not from the function that I made in the parent process. in this situation, execlp was used to overlay a process image that has been created by a call to the fork(). For the pid is greater than 0, wait() function was used that makes parent process to wait for child process to finish.

My hardship was in the windows system, it was hard to compile the code and test weather code is right or wrong because some headers does not work in the windows system. Finally, I figured out that there is different syntax to substitute the other. Also I realized that in Linux, I can compile and everything runs fine. And for me to understand the fork() took me awhile because when I just read and listen about fork(), it wasn’t too hard to understand but making the code out of it was challenge for me.