Unix Assignment6

Team19 108072244 邱煒甯 108032053 陳凱揚 112065525 簡佩如

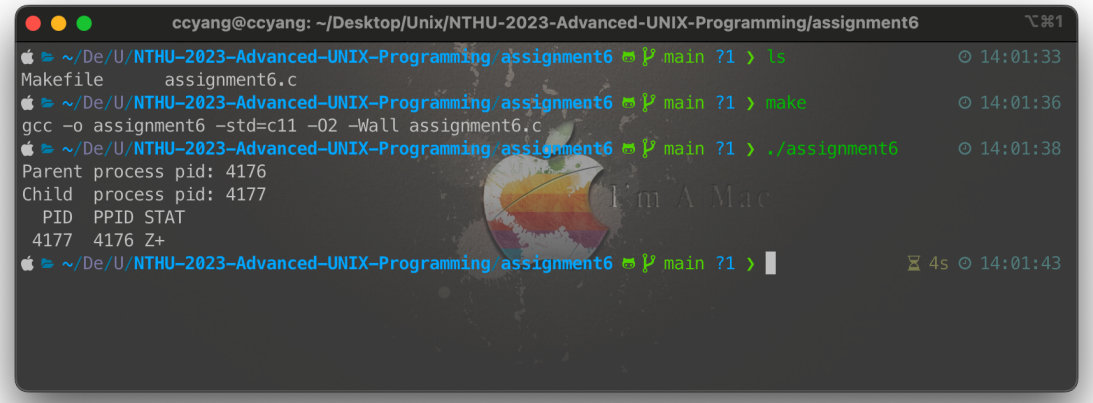
Implementation

```c
    pid_t pid = fork();

    // Check if fork() returns an error
    if (pid < 0) {
        perror("Fork failed");
        exit(1);
    }

    // Child process
    if(pid == 0) {
        printf("Child  process pid: %d\n", getpid());
        exit(0);
    }
    // Parent process
    else {
        printf("Parent process pid: %d\n", getpid());
        // Parent hangs for a while to make sure that child is finished
        // and can be seen as zombie
        sleep(3);
        // Execute 'ps' command, 'Z' state indicates a zombie process
        char command[100] = {};
        sprintf(command, "ps -o pid,ppid,state -p %d", pid);
        system(command);
    }
```

To create a zombie process, the first step is to initiate a child process by invoking the `fork()` function. If the return value of `fork()` is 0, it signifies that the current process is the child process. In this case, we print its PID (Process ID) and promptly exit. On the other hand, if the return value of `fork()` is greater than 0, it indicates that the current process is the parent process, and the value returned is the PID of the newly created child process.

Then, we call the `sleep()` function to hang for a while to make sure that child is finished and can be seen as a zombie process. Afterward, we utilize the `ps` command to inspect the state of the child process. We can see the output of `ps` command show the child process is a zombie process.

Result


ccyang@ccyang: ~/Desktop/Unix/NTHU-2023-Advanced-UNIX-Programming/assignment6
~/De/U/NTHU-2023-Advanced-UNIX-Programming/assignment6 main ?1 ❯ ls 14:01:33
Makefile assignment6.c
~/De/U/NTHU-2023-Advanced-UNIX-Programming/assignment6 main ?1 ❯ make 14:01:36
gcc -o assignment6 -std=c11 -O2 -Wall assignment6.c
~/De/U/NTHU-2023-Advanced-UNIX-Programming/assignment6 main ?1 ❯ ./assignment6 14:01:38
Parent process pid: 4176
Child process pid: 4177
PID PPID STAT
4177 4176 Z+
~/De/U/NTHU-2023-Advanced-UNIX-Programming/assignment6 main ?1 ❯ 4s 14:01:43