

Operating Systems Assignment1 Report

09/18/2016

Jinhao Chen

Project Objective:

The purpose of this project is to introduce us to the concept of shared memory and the problems that can occur if shared memory is not protected adequately. In this assignment, each of processes will share a variable called "total". Each will increment the variable "total" by one to 100,000, 200,000 and 300,000 respectively. After all the children have finished, the parent process should release the shared memory and terminate.

Overview:

In given file ass1.c, it had two processes already; I just create the third process which is third child process. It similar to the other two processes, just make some changes:

```
/* This function increases the value of shared memory variable "total"
 * by one all the way to 300000
 */
void process3()
{
    int k = 0;
    while (k < 300000)
    {
        k++;
        total->value = total->value + 1;
    }
    printf("From process3 total = %d\n", total->value);
}
```

In the main program, the code I added was that one if statement and one while loop. The reason add if statement because we have three process function and we need call all three process. In order to let parent knows precisely when child process has finished, I add a while loop to use the wait function. When child finishes execution, just print the process id of each child.

```
// To run the process3 function
if ((pid1 != 0) && (pid2 != 0) && (pid3 = fork()) == 0)
    process3();

if ((pid1 != 0) && (pid2 != 0) && (pid3 != 0) )
{
    // Use the "wait" function so that the parent knows precisely
    // when each of the children finishes
    while ((kid = wait(NULL)) > 0);
}
```

Conclusion:

I test the program several times, and the follow image is result on Linux. in my opinion, it seems that the result of total number for each process looks incorrect, because when we run the process at the same time in share memory we should get same update for the total number. Therefore this result shows that the problems that can occur if shared memory is not protected adequately.

```
[jinhao@login3 os]$ ./a.out
From process1 total = 100596
From process2 total = 229584
From process3 total = 341146
Child with ID: 15655 has just exited.
Child with ID: 15656 has just exited.
Child with ID: 15657 has just exited.
End of Program.
```

```
[jinhao@login3 os]$ ./a.out
From process1 total = 109166
From process2 total = 254758
From process3 total = 347938
Child with ID: 15672 has just exited.
Child with ID: 15673 has just exited.
Child with ID: 15674 has just exited.
End of Program.
```

```
[jinhao@login3 os]$ ./a.out
From process1 total = 100419
From process2 total = 218726
From process3 total = 324949
Child with ID: 15677 has just exited.
Child with ID: 15678 has just exited.
Child with ID: 15679 has just exited.
End of Program.
```