CS 474 Operating System

Project 2.1 (Shared Memory)

Oct 16, 2022

Project Objectives:

The purpose of this project is to introduce students to the concept of shared memory and the problems that can occur if shared memory is not protected adequately.

Total points Available: 100

Due: Oct 16 at 11:59 pm

Project Description:

In this assignment, you will create **4 child processes**. Each of these processes will share a variable called "total". Each will increment the variable “total” by one to 100,000, 200,000, 300,000 and 500,000 respectively. Make sure that only the newly created child calls the function “process#()”

After **all the children have finished**, the parent process should release the shared memory and terminate. Use the **"wait"** function so that the parent knows precisely when each of the children finishes. The parent should print the process id of each child as the child finishes execution. Then it should release shared memory and print "End of Program".

You need to run program several times and analyze your observations (write report).

Sample output

```
From Process 1: counter = 270547.
From Process 2: counter = 347860.
From Process 3: counter = 400001.
From Process 4: counter = 500000.


Child with ID: 2412 has just exited.
Child with ID: 2411 has just exited.
Child with ID: 2413 has just exited.
Child with ID: 2415 has just exited.

            End of  Simulation.
```

Submitting your assignment

- Submission via Canvas Assignment.
  - It is your responsibility to submit these assignments in a timely fashion.
- All files should be zipped together.
- There should be a readme file explaining in detail the exact steps to be taken to compile and execute the code files and the title page
- Testing of this work should be done only on the CS lab machines. Please make sure these machines are not locked up due to your code. The execution for grading purposes will be done on the lab machines.
- In case of any code errors, partial credit may be offered based on the code and documentation.
- A report that presents the performance evaluation of your solution.
  - The report should be properly formatted (an academic format style, such as ACM or IEEE being preferred) and contain quantitative data along with you analysis of these data.

Development Environment

You may write your program using any available editor Nano, Emacs, Vi or whatever editor you are most comfortable with, BUT, it must compile with gcc and be executable on one of the CS machines.