CS474 Operating System

Project #1 (Shared Memory)

September 17, 2020

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: October 4 at 11:59 pm

Project Description:

In this project, you need to create **4 processes**. Each of these processes will share a variable named "total". Within each process, you need to create a loop and increase the variable "total" **one by one** until 100,000, 200,000, 300,000 and 500,000 respectively. Print out the value of "total" after each process finishes the loop. 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 exactly when each child completes. 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 the program several times and write a report to describe your observations and analyze your results.

Sample output:

From Process 1: Counter = 100001

From Process 2: Counter = 200054

From Process 3: Counter = 300085

From Process 4: Counter = 500000

Child with ID 9099 has just exited.

Child with ID 9100 has just exited.

Child with ID 9098 has just exited.

Child with ID 9101 has just exited.

End of program.

Submitting your assignment

- Each group only need one submission via Canvas Assignment. Include names of your team in the code and report.
- 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 Computer 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 your analysis of these data.

Late Submission Policy

Late work will be not accepted. You have 6 fund/slack days. Use them wisely.

Grading Criteria:

- Minus 90% if code does not compile. Minus 70% if it compiles but does not run.
- If the code compiles and runs, further deductions will be made for the following:
  - Minus 40% if 4 children are not created.
  - Minus 40% if the processes are not cooperating processes.
  - Minus 30% if the children fail to modify the shared variable.
  - Minus 20% if parent ends without waiting for all children to exit.
  - Minus 10% if parent does not release shared memory before ending.
  - Minus 10% if the report is not written
  - Minus 10% if children do not print out their results.
  - Minus 10% if parent does not print each time a child finishes.
  - Minus 5% if no comments.
  - Minus 3% if your name and username is not included in comments on the top of your source code

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 Computer Lab machines.

Compiling a program: gcc -o helloworld helloworld.c
Execute the program: ./helloworld

Enter your account credentials by typing your username and password into the login screen if at a physical lab PC, via PuTTY if at a remote Windows PC, or via *ssh yourusername@hostname.cs.nmsu.edu.* You may go to https://intranet.cs.nmsu.edu/wp/cog/remote-access-to-the-cs-domain/ for detail host list.

To login to these machines remotely, download PUTTY (for Windows, Linux users skip this step) by going to: https://www.chiark.greenend.org.uk/~sgtatham/putty/

Then after the download, execute PUTTY. Their hostnames are *hostname.cs.nmsu.edu*. Then enter login name and password.

Hints:

Build your project in an incremental fashion. Attempt to meet each objective before moving on to the next.

Some useful UNIX commands:

The UNIX commands, needed for the project, are given at the end of lecture 6. Also you can use link

https://beej.us/guide/bgipc/output/html/multipage/shm.html

Find more information about these command and options used by them by using UNIX manual or by simply using man command.