COMP 2560 FALL 2020

Assignment 3

Due date: Nov. 8, 11:59pm

1. Using Unix system calls, *fork(), wait(), read() and write(), write a C program for integer- basic Arithmetics (+, - , *, /). Your program should follow the sequential steps, given below.*
Note terminal keyboard input/output should be done using read()/write() instead of scanf()/printf() in this question.

- Prompts the message "This program makes simple arithmetics",
- Gets in an infinite loop then
    1. Writes the message "Enter an arithmetic statement, e.g., 34 + 132 > ",
    2. Reads the whole input line,
    3. Forks and
        - the parent writes the message "Created a child to make your operation, waiting" then calls wait() to wait for its child.
        - the child process calls the function childFunction(char *) and never returns.
    4. The child, through childFunction(char *line),
        - writes the message "I am a child working for my parent"
        - uses sscanf() to convert the input line into an integer, a character and an integer, respectively.
        - in case of wrong statement, the child calls exit(50)
        - in case of division by zero, the child calls exit(100)
        - in case of a wrong op the child calls exit(200)
        - otherwise, it performs the appropriate arithmetic operation,
        - uses sprint() to create an output buffer consisting of n1 op n2 = result,
        - writes the output buffer to the screen
        - calls exit(0)
    5. Once the child terminates, the parent checks the returned status value using bit-manipulation macros discussed in class and if it is 50, 100 or 200, writes "Wrong statement", "Division by zero" or "Wrong operator", respectively.
    6. The parent goes back to 1.

Important:

- All reads/writes must be done using read()/write()
- You can use the returned value of sscanf() for detecting a "Wrong statement"
- I have attached a file (sscanf_demo.c) to demonstrate typical usage of sscanf(). You could also search online to see how to use it. It is very simple and similar to scanf().

In addition to submitting the source code, please use the script command to record a terminal session with timing log to demonstrate that you compile the program and run the program. Submit the script files as well.