Project 1 Programming – Time Being Used **CS 300, Hong**

All the coding must be done in C language, being able to compile and execute in cs-intro.ua.edu. If you use a different OS environment in completing the project, you should take extra care to test your C code in cs-intro.ua.edu so to make sure it can compile and run as you expected.

Programming Project 1, Time Being Used

The description can be found in the attached PDF file ***P1description_timeused.pdf.*** Read carefully. The project is clearly described in the PDF file.

Additional project requirements:

0. Source code should be written in C language
1. Add to your program the code of getting process IDs for both parent and child processes, and output the PIDs at appropriate places with the format of "Parent PID: xxx" or "Child PID: yyy".
2. Software: at the beginning of your source code, include comment lines of your full name and CWID. Other comment lines are optional, but recommend. Name source code file as: *Fullname_time.c*
3. Write a short report (around 2 pages) on this project, including screenshots of the output and a brief explanation of your understanding on the flow of the program, and the outputs from different commands. Name your report as: *Fullname_time_report.doc (or pdf).*

Compiling and testing

1. use `execvp( );`
2. Compile your source code using `gcc -Wall filename -std=c99 -o time.` That will produce an executable code `time` in the current directory. Run the program by using `./time` *`command`*
3. *`command`* can be any Linux commands. For examples: cat filename, vim filename, ls -lt, etc..
4. To test the time output, you can use a few files of different sizes.

Submission and grading:

1. Submit to Blackboard "Proj 1" before deadline. Submit only the source code, and the report. do NOT zip.
2. Grading uses compilation command `gcc -Wall filename -std=c99 -o time` at the **cs-intro.ua.edu** server.
3. If you develop your program at your local machine, you should try to compile your source code using the same command at the cs-intro server so to avoid compilation errors at grading time, which, will lead to points drop.

Delay policy:

Each one-day delay will drop 20% points. You should start your projects early to avoid potential last minute issues that cause late turn-in.

Grading will count:

a) submission and compilation, 10 points;
b) modified main, gettimeofday() and time calculation;
c) use of fork() and wait(), and execvp();
d) use of shared memory and shm_open(), mmap(), and shm_unlink();
e) testing of various commands;

FAQs:

(1) As long as it compiles, i.e., produces an executable file, it is ok.
(2) When grading, only valid commends will be tested.
(3) Test your program at cs-intro.
(4) Email your question to TA or me, join office hours.