Project 1

Submit Assignment

Due Feb 14 by 11:59pm **Points** 100 **Submitting** a file upload **Available** until Feb 17 at 11:59pm

Do **Project 1 – UNIX** Shell in textbook Chapter 3's Programming Projects Section. The project must be written in C. Your program must compile and run on pyrite.

Requirements

makefile (5 points)

You get 5 points for using a makefile. Name your source files whatever you like. Please name your executable **osh**. Be sure that "make" will build your executable.

Documentation (10 points)

If you have more than one source file, then you must submit a **Readme** file containing a brief explanation of the functionality of each source file. Each source file must also be well-documented. There should be a paragraph or so at the beginning of each source file describing the code; functions and data structures should also be explained. Basically, another programmer should be able to understand your code from the comments.

Main Loop (10 points)

The structure of the main loop is given in the project description. You program should terminate when the user types the command "stop".

Executing Command in a Child Process (20 points)

A "foreground job" is a command of the form ARG0 ARG1 ... ARGn .

Examples:

cp file1 file2

cp -R dir1 dir2

rm -f -R dir1 dir2 file1 file2

A foreground job should be executed as follows:

1. Create a new process using fork()
2. Replace the image of the created process with the desired command using execvp().
3. Wait for the process to terminate using waitpid().

A "background job" is a command of the form ARG0 ARG1 ... ARGn & .

Examples:

cp file1 file2 &

cp -R dir1 dir2 &

Your shell does not need to support background jobs. That is, you can assume that the user never adds an '&' at the end of a command.

Providing a History Feature (15 points)

See project description for details

Redirecting Input and Output (20 points)

See project description for details

Communication via a Pipe (20 points)

See project description for details

Project Submission

Put all your source files (including the makefile and the README file) in a folder. Then use command zip -r <your ISU Net-ID> <src_folder> to create a .zip file. For example, if your Net-ID is ksmith and project1 is the name of the folder that contains all your source files, then you will type zip -r ksmith project1 to create a file named ksmith.zip and then submit this file.