

# COSC 6360-Operating Systems

## Assignment #1: A Simple Shell

New Due Date **October 8, 2020 anywhere on earth.**

### OBJECTIVE

This assignment should help you to understand the functions of a command language interpreter and learn how to create processes and process pipes.

### SPECIFICATIONS

Your program should prompt for input from its standard input. (Use the prompt “**MyShell:**” to avoid confusions.) Every line of input will consist of a sequence of arguments separated by an arbitrary number of spaces. The first of these arguments will either be the name of a program to be executed or the keyword “**redirect**”. Any process whose command line starts with the keyword “**redirect**” should have its standard output redirected to the process following the keyword “**into**” as in:

**MyShell: redirect ls into more**

The result should be identical to that of the standard pipe shell construct

**\$ ls | more**

In addition, your shell should handle internally two process control commands:

1. **newpath**, which takes as arguments a list of directories (separated by colons) and makes this list the new list of directories to be searched for executables. It should replace the list of directories specified by the PATH your shell inherited from its parent.
2. **exit**, which terminates the shell.

Your shell should issue error messages any time it encounters:

1. A process that **execvp()** cannot find.
2. The keyword “**redirect**” in first position without the keyword “**into**” or with that keyword in the last position.

You are not allowed to use any of the system calls **system()**, **execv()** or **execve()**.

### SPECIFICATIONS

- Before you start your assignment, familiarize yourself with the UNIX system calls **fork()**, **wait()**, **execvp()**, **dup()**, **pipe()**, **exit()**, **\_exit()**, and **setenv()** commands.

- You do not have to treat the special characters and metacharacters of **/bin/sh** and **/bin/bash** in any special way.
- Do not worry about signals.
- You will find that the command **cd** will not work properly (but the fix is very simple). Remember that UNIX functions normally return a zero value when they are successful.

### WRITING YOUR PROGRAM ON YOUR COMPUTER

The best way of experimenting the full power of Linux is to develop your program on *your own* Linux machine, virtual or real. When this is not possible, there are easier alternatives:

- If you have a Mac, there is nothing to install, as the macOS bash shell is a Unix/Linux shell.
- If you have a PC with an up to date 64-bit version of Windows 10, you *should* install a Linux shell. The best guide on how to do it comes straight from Microsoft:

[https://msdn.microsoft.com/en-us/commandline/wsl/install\\_guide](https://msdn.microsoft.com/en-us/commandline/wsl/install_guide)

- If your computer runs any other version of Windows, your best bet is to install Cygwin from [cygwin.com](http://cygwin.com). You can find detailed instructions in:

<https://www.slothparadise.com/how-to-install-cygwin-on-windows/>

The tool is well integrated with other Windows apps, but does not always emulate correctly all Linux system calls.

- Do not be afraid to ask for instructor's help.

### SUBMISSION PROCEDURE

*Once you have submitted your assignment through Blackboard, you must schedule an appointment with the instructor to get a chance to explain how your program works. **This is the only way you will get your assignment graded.***

This document was updated last on Friday, October 2, 2020.