Project 3: The Shell

This project was written by instructors of CS354 at Purdue University, and is taken from their web site.

Second part: Process Creation, Execution, File Redirection, Pipes, and Background

Starting from the command table produced in Part 1, in this part you will execute the simple commands, do the file redirection, piping and if necessary wait for the commands to end.

1. For every simple command create a new process using *fork()* and call *execvp()* to execute the corresponding executable. If the _bakground flag in the Command struct is not set then your shell has to wait for the last simple command to finish using *waitpid()*. Check the manual pages of *fork()*, *execvp()*, and *waitpid()*. Also there is an example file that executes processes and does redirection in cat_grep.cc. After this part is done you have to be able to execute commands like:

   ```
   ls -al
   ls -al /etc &
   ```

2. Now do the file redirection. If any of the input/output/error is different than 0 in the Command struct, then create the files, and use *dup2()* to redirect file descriptors 0, 1, or 2 to the new files. See the example cat_grep.cc to see how to do redirection. After this part you have to be able to execute commands like:

   ```
   ls -al > out
   cat out
   ls /tttt >& err
   cat err
   cat < out
   cat < out > out2
   cat out2

   ls /tt >>& out2
   ```

   ```
   ">& file" redirects both stdout and stderr to file. ">>& file" append both
   stdout and stderr to file.

   ">> file" appends stdout to file.
   ```

3. Now do the pipes. Use the call *pipe()* to create pipes that will interconnect the output of one simple command to the input of the next simple command. use *dup2()* to do the redirection. See the example cat_grep.cc to see how to construct pipes and do redirection. After this part you have to be able to execute commands like:

   ```
   ls -al | grep command
   ls -al | grep command | grep command.o
   ls -al | grep command
   ls -al | grep command | grep command.o > out
   cat out
   ```

   The deadline of this part of the project is Wednesday, April 2, 2003 at 11:59pm.

   Follow these instructions to turnin your part two.

   1. Login to CSSUN.

2. cd to lab3-src and type "make clean"

3. Type "make" to make sure that your shell is build correctly.

4. Type "make clean" again.

5. cd one directory above lab3-src

6. Create a tar file named <user_name>.tar, where <user_name> is your CSSUN login, by typing

```
tar -cf <user_name>.tar lab3-src
```

7. Gzip the tar file by typing

```
gzip <user_name>.tar
```

8. Since this timestamp will be used to verify whether the work was completed on time or not, you should set the permissions on the file you submitted to make sure that the file timestamp is not changed. So this by typing:

```
chmod a-w <user_name>.tar.gz
```

9. Mail the gzipped tar file to clay at cs dot georgetown dot edu as an attachment.

Testing your Shell:

Your shell will be graded using automatic testing, so make sure that the tests given to you run.

Your grade for this project will depend on the number of tests that pass. The tests given are for part 2 and 3 of the project. Tests used for grading: test-shell-grading/ and test-shell-grading.tar.Z

See the file lab3-src/README for an explanation on how to run the tests. The tests will also give you an estimated grade. This grade is just an approximation. Other tests not given to you will be used as well during grading.