CSI 3336 Systems Programming
Homework 1

Playing around on Earth

You will have a chance to get familiar using the command-line interface offered by the shell and using your chosen text editor. Even if you have a Linux machine of your own, you will need to complete this assignment on one of the department's general-purpose Linux systems. You will turn in a series of printouts for this assignment. When you redirect output to a .txt file, print out the contents of that file for submittal.

Log on to the Linux systems using putty, or some other secure shell client. First, change your password using the password command. When your account was created, it was assigned a randomly-generated password. You will want to change this to something that's easy for you to remember but hard for anyone else to guess. Just type the command passwd at the shell prompt. This will prompt you for your old password and give you a chance to choose a new password. This change will automatically alter your password on all of our Linux machines.

I have created a cs3336/a1 directory in my "maars" account. Create a "cs3336/a1" subdirectory in your system, and copy **everything** (files and directories) in my "cs3336/a1" to your "cs3336/a1" directory in Linux. Do a long list of everything in this subdirectory of your account and redirect the output to a1out0.txt.

Look around the directories and subdirectories in cs3336/a1. You have to perform a few operations on files in these directories.

1. If you find any files with names ending in ".junk", delete them. Likewise, if you find any directories with names ending in ".junk", delete them and anything they contain. Do a long list of everything in this subdirectory and redirect the output to a1out1.txt.

2. If you find any hidden files, rename them to a similar name that's not a hidden file using the rename command. Do a long list of everything in this subdirectory of your account and redirect the output to a1out2.txt.

3. Make a directory called cs3336/a1/keep. Look in all the files in cs3336/a1 and move any file that contains the word "important" on the first line to cs3336/a1/keep. You can look at the contents of a text file using your favorite editor or simply by using cat or more (You can learn more about most commands by using "man" to display the manual for a specific command. E.g. man cat will provide you with the manual for the command cat. You may even try playing around with a useful command named grep). Do a long list of everything in cs3336/a1 and redirect the output to a1out3.txt. Then, do a long list of everything in cs3336/a1/keep and append the output to a1out3.txt.

4. You will find a few source files scattered around. These are the files ending in ".c". You should compile them and leave a corresponding executable file in the same directory. Use your text editor (you may want to skip to the end of this assignment and work through the tutorial of your chosen editor first) to add appropriate comments to the programs and to fix any errors in them making sure to retain the original intended functionality. Remember, that some versions of the C compiler will expect C-style comments, /*...*/, not the //... form of C++. Make sure that you review the coding conventions document found on canvas for the classes format for proper commenting. The following command can be used to compile a source file, using c89 standard, called "work.c" and produce an executable named "work".

```
gcc -ansi -pedantic -Wall -o work work.c
```

Print out the contents of any files you edit, and turn them in with this submittal. Also, do a long list of this directory and redirect the output to a1out4.txt.

5. See who's on the system while you're working and save this output to a file named a1out5.txt. You can do this by redirecting the output of the who command.

Meet your Editor

Getting familiar with your text editor is one of the most useful things you can do right now. Doing the following should help.

1. My "maars/cs3336/a1" directory contains a file named .vimrc. Put a copy of the file in your home directory. The .vimrc file contains a few helpful customizations for vim users.

2. Work through the tutorial for your editor, run vimtutor.

3. Next, use your editor to make a .plan file. If you have a file named .plan in your home directory, its contents will be printed out when someone fingers your user id. The original intent was that these files could tell others what you are working on. In practice, people just use them to report information about themselves or show an ASCII art picture. The system worked kind of like a primitive home page. You can test the functionality by typing finger mars - where maars is the username you wish to see information about. Make a file named .plan in your home directory and put some useful information about yourself that you don't mind sharing with others. You should be able to view your plan file when you finger your user id. However, the default permissions won't let others see it by default. Type chmod +r .plan and <enter> (this will add read permissions to the user, group, and others, allowing them to read your .plan file once they have access to your home directory). In addition you will need to add an additional permission to your home directory. Type chmod +x ../ $USER (this will add execute permissions to the user, group, and others to your home directory which allows them access to your home directory)