Assignment 1

September 16, 1999

Question 1 (10 Marks)

Write a csh script called "**checkmail**" that checks for new mail messages every 60 seconds, and prints a message when new messages have been detected. Your script can do this by monitoring the size of the file /var/spool/mail/<username>, and assuming that anytime this file <u>increases</u> in size new messages have arrived. For example, if a user with a login name **maclean** runs the script, it should monitor the file /var/spool/mail/maclean. If another user runs the same script, it must check their file instead. Your script should report an error if the file does not exist.

Hand in a printed version of your csh script, as well as submitting it electronically on CDF using "submit -N a1 csc209h checkmail". You can overwrite a previous submission by adding the "-f " switch to the submit command.

Question 2 (5 Marks)

From the course website, download the file Broken.c. This C program has both compile-time and run-time errors in it. Make a copy of the file named Fixed.c, and find and correct the errors (be sure to read the instructions included at the start or Broken.c).

Submit the corrected program using the command "submit -N a1 csc209h Fixed.c". Also, hand in a printed version of the corrected program.

Question 3 (15 Marks)

Write a csh script named "checkLoad" to print the name of the computer with the lightest "load". The load is a number indicating how busy a given computer is. The "ruptime" program gives the current load of a list of computers available on the network, as shown below. Use the second of the three load numbers to make decisions in your script. A load of 0.0 is thus a light load, while one of 3.0 would be a heavy load. If two computers have an equal load, the one with fewer users should be assumed to be the better.

```
"ruptime" output:
  agrajag.cdf     down 26+02:17
  amber.cdf         up  1+07:07,     2 users,   load 0.00, 0.00, 0.01
  amethyst.cdf      up    22:46,     3 users,   load 0.04, 0.03, 0.07
  anjie.cdf       down  1+09:55
  axiomset.cdf      up  1+07:49,     2 users,   load 0.05, 0.03, 0.03
  babel.cdf       down  1+09:57
  benjy.cdf       down  1+09:54
  clique.cdf        up  1+02:25,     3 users,   load 0.07, 0.07, 0.05
```

Here are some hints: "grep" may be used to exclude lines containing a particular pattern (the man page will tell you how); "sort +n" will perform an ascending sort of lines based upon field n+1.

Hand in a printed version of your csh script, as well as submitting it electronically on CDF using "submit -N a1 csc209h loadCheck". You can overwrite a previous submission by adding the "-f " switch to the submit command.

Question 4 (10 Marks)

Write a csh script named "recent" which accepts a program name as an argument and lists all programs by that name found in the directories listed in the "`path`" shell variable. At the end, the shell script should execute the first instance of the program that it found, and print the status code returned by that program. You should do basic error checking. The `path` shell variable contains the list of all directories to search for executable programs, separated by spaces, *e.g.*:

```
echo $path
/u/maclean/bin /local/bin /local/bin/X11
```

For example, if there was a program 'ls' in both /u/maclean/bin and /local/bin, the output of the command "recent ls" would be:

```
/u/maclean/bin/ls
/local/bin/ls
Executing /u/maclean/bin/ls
    (output of ls will be here)
Returned status code: 0
```

Hand in a printed version of your csh script, as well as submitting it electronically on CDF using "`submit -N a1 csc209h recent`". You can overwrite a previous submission by adding the "-f " switch to the submit command.