# CS205 - LAB1 - Extra Credit

You have seen during the lecture that "switch" doesn't allow using strings.

It's very frequent to have programs that control other programs that run as services (we'll talk about programs that run as services and how a program can control another program towards the end of the course). For instance, there is a famous Web server program called "apache" and when you want to control it you start another program called "apache\_ctl".

This program (just like a console) displays a prompt such as "> " and then expects you to type a command, for instance "start" to start the server, "stop" to stop it, "restart" to stop it first and then start it, "reload" to make it read again a configuration file that was modified, etc. Of course there is also an "exit" command to quit apache\_ctl.

You are asked to write a program that will accept the following commands: start, stop, restart, status, exit (to quit it) and, when a command other than "exit" is recognized, will simply display "command <name here> recognized". It must also say "Invalid command" if the command isn't recognized.

You are asked to test the command in a switch statement, not an if ... else if ... else if ... structure.

For this, you'll have an array of strings containing the commands, you'll search it, and if you find the command you'll return its index in the array. The index is an integer and can be used for "switch".

For legibility, you'll associate a symbol to each index.

For instance if you have

char \*commands[] = {"start", "stop", ... };

You can have

#define START\_CMD 0

#define STOP\_CMD 1

and use in the switch:

case START\_CMD: ...