Week 3 Lab

This Week

The backlog for this week introduces functions and begins your transition to C. It covers two weeks. This week you should be prepared to demo the Scratch programs; the following week you should be prepared to demo the C programs.

C Programs are written using the Linux (Unix) operating system. It is different from the Window operating system, which you are probably using now. To prepare for working in Linux, do the following exercises in class. If you cannot complete the tasks this week, you may continue next week. After you have completed these tasks you will have completed the Lab for week 4.

1. Log on to Linux
2. Open a terminal.
3. See which directory your are in.
4. Create a new directory in your home directory called “test”
5. List the directories in your home directory.
6. Move into the “test” directory
7. Create a new directory in the “test” directory called “temp”
8. Try to remove the new directory using “rm.” What happens?
9. Create a new file called “readme” using the “cat” command that contains the words “Hello World”.
10. List the “readme” file using the “cat” command.
11. List the “readme” file using the “less” command.
12. List the contents of the “test” directory showing permissions.
13. Change the permission of “readme” to disallow world read and write and group write.
14. List again showing permission. What changed?
15. List the contents of the “test” directory showing permission and files that start with a “.”
16. Delete the “readme” file.

17. Move back to your home directory.
18. Try to remove the “test” directory. What happens.
19. Remove the “test” directory without going into the “temp” directory.
20. Remove the “temp” directory.
21. Move to the root directory.
22. List the contents to the root directory showing permissions and files that start with a “.”
23. List the content of the home directory. Find your home directory and list the contents of it.
24. Try to remove the “home” directory. What happens?
25. Create a directory called project1 for the Hello World project
26. Write “helloWorld.c” using nano

```
#include <stdio.h>

int main(int argc, char *argv[])
{
        printf("Hello world\n");
        return 0;
}
hello.c (END)
```

```
all: hello

hello: hello.o
        gcc -o hello hello.o

hello.o: hello.c
        gcc -c hello.c

clean:
        rm hello.o hello
Makefile (END)
```

27. 27. Compile hello world using gcc

Demo

Today you will demo the backlog from last week.

Marks

- Story 1 (15 Marks)
- Story 2 (15 Marks)