CS 271 Lab 1 - C Basics

Goals:

- **Learn to use Kate editor to create and debug C programs.**
- **Learn to use gcc compiler with math library and one source file.**
- **Learn to convert Java syntax to C syntax**
- **Learn Linux commands: cd, pwd, cp, dir, mkdir, mv, and rm**

1. Log in. Open the Kate editor.

   To make the Terminal window visible in Kate, do the following, go to the Settings menu.
   ➔ Configure Kate
   ➔ Plugins
   ➔ check the box next to Terminal Tool View
   ➔ then click OK.

Linux Commands

pwd	"present working directory" – shows the full path of the current directory Note: a folder and a directory are the same thing.
cd ___	"change directory" – changes the working directory to the directory specified in the command. cd with nothing after it changes to your home directory. You can specify a full path or just a subdirectory name. Examples: cd changes to home directory cd cs271 changes to the subdirectory named cs271 cd .. (two periods) changes to one directory higher than the current one cd /home/cs271/Lab1 uses full path
cp file1 file2	"copy" -- copies file1 to file2. **Caution!** If file2 already exists, it is overwritten.
mv file1 file2	"move" – Linux command for rename. Renames file1 to file2.
rm file	"remove" – Linux command for delete.
dir	"directory" – displays a list of files in the current directory. The * may be used as a wildcard. Example: dir *.c -- displays all files that end with .c
mkdir ___	"make directory" – creates a new directory

2. **Terminal Window:** Click in the terminal window. Type **`pwd`** and press Enter. Note the directory information. Type **`cd`** and press Enter. Type **`pwd`** again and note the difference in the directory information.

3. **Terminal:** Type **`mkdir cs271`** and press Enter. Change directory to cs271 by typing **`cd cs271`**. Type **`mkdir Lab1`** and press Enter. Type **`cd Lab1`** to change directory to Lab1. This is where you will save your files

4. **Editor:** Type in the first program from the Lab 1 handout given in class. Don't have the handout?? Visit the bulletin board at Science Hall room 157 and get a copy. Don't borrow someone else's copy.

   Save the file as program1.c in the cs271/Lab1 directory.

Reminder: In the terminal window, you must set your working directory to the folder where you saved your program.

5. **Terminal:** In the Terminal window, compile the program and produce an executable file named program1. Here's the command:

```
gcc program1.c -o program1
```

   If you have syntax errors, go back to the editor window and fix them.

6. **Terminal:** Run the program by typing a period, a slash, then the name of the executable:

```
./program1
```

   The program should prompt you for two integers then display the sum of the integers. If it doesn't, you have some debugging to do.

7. **Editor:** Close program1.c.

8. **Terminal:** Copy program1.c to a new file named circle.c. The command is:

```
cp program1.c  circle.c
```

9. **Editor:** Open circle.c. Change the program so that it will do the equivalent of the Java program given in the handout. Don't forget to change the header comments, also.

   Save the file.

10. **Terminal:** Compile the program with:

```
gcc circle.c –lm -o circle
```

Note: -lm (dash el em) is required for programs that use math.h

   Debug if needed.

11. **Terminal:** Run the program several times with different input values to make sure that you've thoroughly tested all of the statements. Debug if needed.

```
./circle
```

12. **Editor:** Create a new program. *((Try to type in the complete program without looking at the book or previous programs.))* Save it as hypotenuse.c.

   Write a C program to perform the tasks shown in the "Pseudocode to Implement in C" from the lab 1 handout.

13. **Terminal:** Compile, debug, and test the program.

Submit program1.c, circle.c, and hypotenuse.c on Canvas. (do not tar or zip files)

Before the next lab:

- **Speed up your typing. Here are some web sites: www.powertyping.com, www.typingtest.com, www.typing-lessons.org, www.typeonline.co.uk, http://games.yahoo.com/console/tps**
- **Do the self-review exercises at the end of chapter 2**
- **Write additional programs. I recommend exercises 2.24, 2.26, 2.30 and 2.32.**