Experiment 3: C programs using System calls and C libraries

1. Write a program to print version number of your Linux OS and further hardware details ie. sysname, release, version, processor using *Uname* system call.
2. Write a program to print the system statics system uptime(hour, minute, second), total ram space, free ram space, process count, page size , size of user memory using *sysinfo* stub function.
3. Write a program to check the access permission of a given file towards the current process using *access* system call. [ensure the file existence then check the given file has read/write/execute permissions]
4. Write a program to print the current time and date of the day using
   a) *gettimeofday* system call
   b) *ctime lowlevel* library function
5. Write a program to implement *cp* command in linux using
   a) *open – create – read – write* system calls
   b) *open – fstat – sendfile* system calls
6. Write a program to implement *ls* command (basic functionality) using *scandir* low level functions. Modify the same using *stat* system call to incorporate *ls –l* functionality.
7. Implement *du*(disk usage) command using *fstat* system call.
8. Write a program to determine the file owner's write permissions using *stat* system call
9. Implement a program to grep dictionary (which search for a word in the dictionary using "*system* function". Through this function basically you might be executing "*grep –x* string filename" shell command.
10. Write a program to print out the current process's user and system time using *rusage* system call.

References :

1) GNU C Library

http://www.gnu.org/software/libc/manual/html_mono/libc.html#Introduction

2. Advanced Linux Programming : Mark Mitchell, Jeffrey Oldham and Alex Samuel; [for sample programs]
3. Linux kernel Development : Robert Love [Chapter 5]