

Operating Systems Lab 2: Working with Directory Structures

1. Write a C program to emulate the `ls -l` UNIX command that prints all files in a current directory and lists access privileges, etc. DO NOT simply exec `ls -l` from the program.

Code:

```
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
#include <time.h>
#include <string.h>
#include <pwd.h>
#include <grp.h>
#include <locale.h>
#include <langinfo.h>
#include <stdint.h>

char const *perms(__mode_t mode){
    static char local_buff[16] = {0};
    int i = 0;
    //user permissions
    local_buff[i++] = ((mode & S_IRUSR) ? 'r' : '-');
    local_buff[i++] = ((mode & S_IWUSR) ? 'w' : '-');
    local_buff[i++] = ((mode & S_IXUSR) ? 'x' : '-');
    //group permissions
    local_buff[i++] = ((mode & S_IRGRP) ? 'r' : '-');
    local_buff[i++] = ((mode & S_IWGRP) ? 'w' : '-');
    local_buff[i++] = ((mode & S_IXGRP) ? 'x' : '-');
    //other permissions
    local_buff[i++] = ((mode & S_IROTH) ? 'r' : '-');
    local_buff[i++] = ((mode & S_IWOTH) ? 'w' : '-');
    local_buff[i++] = ((mode & S_IXOTH) ? 'x' : '-');

    return local_buff;
}

int main(int argc, char *argv[]){
    DIR *mydir;
    struct dirent *myfile;
    struct stat mystat;
    if(argc == 1)
        mydir = opendir(".");
    else
        mydir = opendir(argv[1]);
```

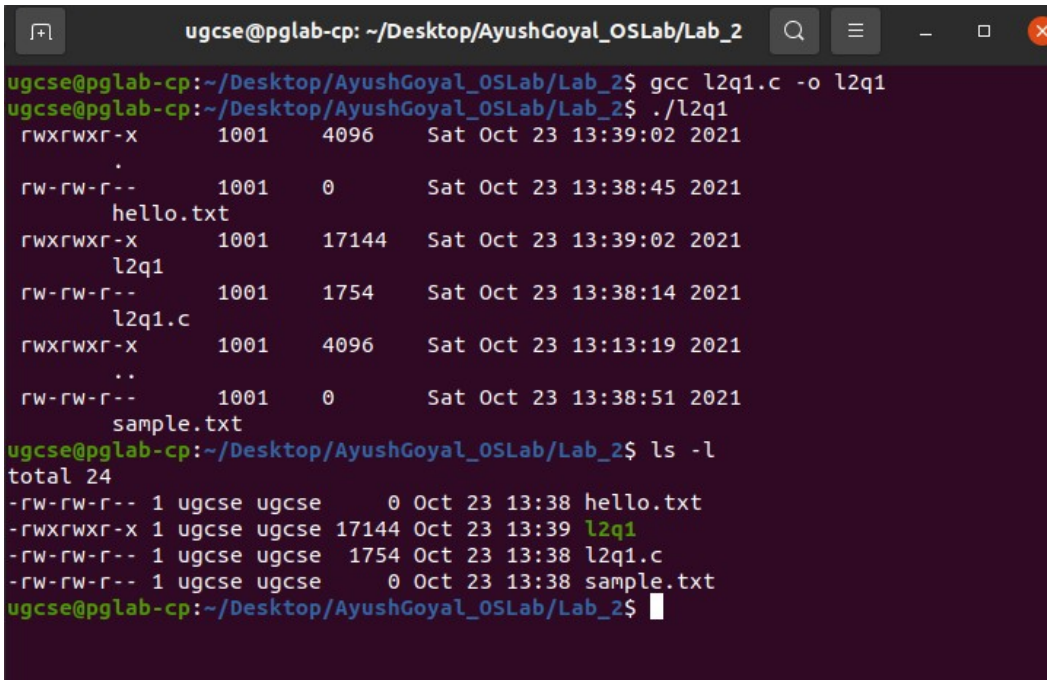
```

while((myfile = readdir(mydir)) != NULL){
    bzero(&mystat, sizeof(mystat));
    stat(myfile->d_name, &mystat);
    printf("%10.10s", perms(mystat.st_mode));
    printf("\t%d", mystat.st_uid);
    printf("\t%ld\t",mystat.st_size);
    printf("%s\t", ctime(&mystat.st_mtime));
    printf("%s\n", myfile->d_name);
}
closedir(mydir);
}

```

The output of the above executed program resembles that of the UNIX command `ls -l` as shown in the screenshot below:

Output:



```

ugcse@pglab-cp: ~/Desktop/AyushGoyal_OSLab/Lab_2
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ gcc l2q1.c -o l2q1
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ ./l2q1
-rwxrwxr-x    1001    4096   Sat Oct 23 13:39:02 2021
.
-rw-rw-r--    1001     0     Sat Oct 23 13:38:45 2021
hello.txt
-rwxrwxr-x    1001   17144   Sat Oct 23 13:39:02 2021
l2q1
-rw-rw-r--    1001   1754    Sat Oct 23 13:38:14 2021
l2q1.c
-rwxrwxr-x    1001    4096   Sat Oct 23 13:13:19 2021
..
-rw-rw-r--    1001     0     Sat Oct 23 13:38:51 2021
sample.txt
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ ls -l
total 24
-rw-rw-r-- 1 ugcse ugcse    0 Oct 23 13:38 hello.txt
-rwxrwxr-x 1 ugcse ugcse 17144 Oct 23 13:39 l2q1
-rw-rw-r-- 1 ugcse ugcse  1754 Oct 23 13:38 l2q1.c
-rw-rw-r-- 1 ugcse ugcse    0 Oct 23 13:38 sample.txt
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$

```

2. Write a program that will list all files in a current directory and all files in subsequent subdirectories.

Code:

```

#include<unistd.h>
#include<sys/stat.h>
#include<sys/types.h>
#include<stdio.h>
#include<stdlib.h>
#include<dirent.h>
#include<string.h>

void executeFunc(char* dir, int depth){

```

```

struct dirent *myfile;
struct stat mystat;
DIR *mydir = opendir(dir);
chdir(dir);
while((myfile = readdir(mydir)) != NULL){
    bzero(&mystat, sizeof(mystat));
    stat(myfile->d_name, &mystat);
    for(int i=0;i<depth;i++)
        printf(" ");
    printf("%s", myfile->d_name);
    if(S_ISDIR(mystat.st_mode))
        printf("/");
    printf("\n");
    if(S_ISDIR(mystat.st_mode) && strcmp(myfile->d_name, ".") != 0 &&
strcmp(myfile->d_name, "..") != 0){
        executeFunc(myfile->d_name, depth+1);
    }
}
chdir("..");
closedir(mydir);
}

int main(){
    executeFunc(".", 0);
    return 0;
}

```

I have created two directories “testdir” and “newdir” to test and added text files in them to test the above program. The output shown below lists all files and directories as required:

Output:

```

ugcse@pglab-cp: ~/Desktop/AyushGoyal_OSLab/Lab_2
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ gcc l2q2.c -o l2q2
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ ./l2q2
./
testdir/
./
test1.txt
test2.txt
../
l2q2.c
hello.txt
l2q1
l2q2
l2q1.c
../
sample.txt
newdir/
./
new2.txt
new1.txt
../
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ ls
hello.txt l2q1 l2q1.c l2q2 l2q2.c newdir sample.txt testdir
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$

```

3. How do you list all installed programs in Linux?

Code:

```
$ dpkg -l
```

Output:

```
ugcse@pglab-cp: ~/Desktop/AyushGoyal_OSLab/Lab_2
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ dpkg -l
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/half-f-inst/Trig-await/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
++-----+
|| Name                      Version                      Architecture Description
++-----+
|| accountsservice           0.6.55-0ubuntu12-20.04.4    amd64         query and manipulate user account information
|| accountsservice-ubuntu-schemas 0.0.7+17.10.20170922-0ubuntu1 all           AccountsService schemas for Ubuntu
|| acl                        2.2.53-6                    amd64         access control list - utilities
|| acpi-support               0.143                       amd64         scripts for handling many ACPI events
|| acpid                      1:2.0.32-1ubuntu1           amd64         Advanced Configuration and Power Interface event daemon
|| adduser                    3.118ubuntu2                all           add and remove users and groups
|| adwaita-icon-theme         3.36.1-2ubuntu0.20.04.2     all           default icon theme of GNOME (small subset)
|| aisleriot                  1:3.22.9-1                  amd64         GNOME solitaire card game collection
|| alsa-base                  1.0.25+dfsg-0ubuntu5         all           ALSA driver configuration files
|| alsa-topology-conf         1.2.2-1                     all           ALSA topology configuration files
|| alsa-ucm-conf              1.2.2-1ubuntu0.5            all           ALSA Use Case Manager configuration files
|| alsa-utils                 1.2.2-1ubuntu2              amd64         Utilities for configuring and using ALSA
|| amd64-microcode            3.20191218.1ubuntu1         amd64         Processor microcode firmware for AMD CPUs
|| anacron                    2.3-29                      amd64         cron-like program that doesn't go by time
|| apg                         2.2.3.dfsg.1-5              amd64         Automated Password Generator - Standalone version
|| app-install-data-partner    19.04                       all           Application Installer (data files for partner applications/repositories)
|| apparmor                   2.13.3-7ubuntu5.1           amd64         user-space parser utility for AppArmor
|| apport                      2.20.11-0ubuntu27.20        all           automatically generate crash reports for debugging
|| apport-gtk                 2.20.11-0ubuntu27.20        all           GTK+ frontend for the apport crash report system
|| apport-symptoms            0.23                        all           symptom scripts for apport
|| appstream                  0.12.10-2                   amd64         Software component metadata management
|| apt                         2.0.4                        amd64         commandline package manager
|| apt-config-icons           0.12.10-2                   all           APT configuration snippet to enable icon downloads
|| apt-config-icons-hidpi     0.12.10-2                   all           APT configuration snippet to enable HiDPI icon downloads
|| apt-transport-https        2.0.4                        all           transitional package for https support
|| apt-utils                  2.0.4                        amd64         package management related utility programs
|| aptdaemon                  1.1.1+bzr982-0ubuntu32.3     all           transaction based package management service
|| aptdaemon-data             1.1.1+bzr982-0ubuntu32.3     all           data files for clients
|| apturl                     0.5.2ubuntu19               amd64         install packages using the apt protocol - GTK+ frontend
|| apturl-common              0.5.2ubuntu19               amd64         install packages using the apt protocol - common data
|| aspell                     0.60.8-1ubuntu0.1           amd64         GNU Aspell spell-checker
|| aspell-en                   2018.04.16-0-1              all           English dictionary for GNU Aspell
|| at-spi2-core                2.36.0-2                    amd64         Assistive Technology Service Provider Interface (dbus core)
|| avahi-autotpd               0.7-4ubuntu7.1              amd64         Avahi IPv4LL network address configuration daemon
|| avahi-daemon                0.7-4ubuntu7.1              amd64         Avahi mDNS/DNS-SD daemon
|| avahi-utils                 0.7-4ubuntu7.1              amd64         Avahi browsing, publishing and discovery utilities
|| baobab                      3.34.0-1                    amd64         GNOME disk usage analyzer
|| base-files                  11ubuntu5.3                 amd64         Debian base system miscellaneous files
|| base-passwd                 3.5.47                       amd64         Debian base system master password and group files
lines 1-44
```

4. How do you find out what RPM packages are installed on Linux?

Code:

```
$ rpm -qa
```

RPM is not installed on Ubuntu by default. Can be installed by following command:

```
$ sudo apt install rpm
```

Output:

```
ugcse@pglab-cp: ~/Desktop/AyushGoyal_OSLab/Lab_2
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$ rpm -qa
Command 'rpm' not found, but can be installed with:
sudo apt install rpm
ugcse@pglab-cp:~/Desktop/AyushGoyal_OSLab/Lab_2$
```

THE END