AUP LAB-2

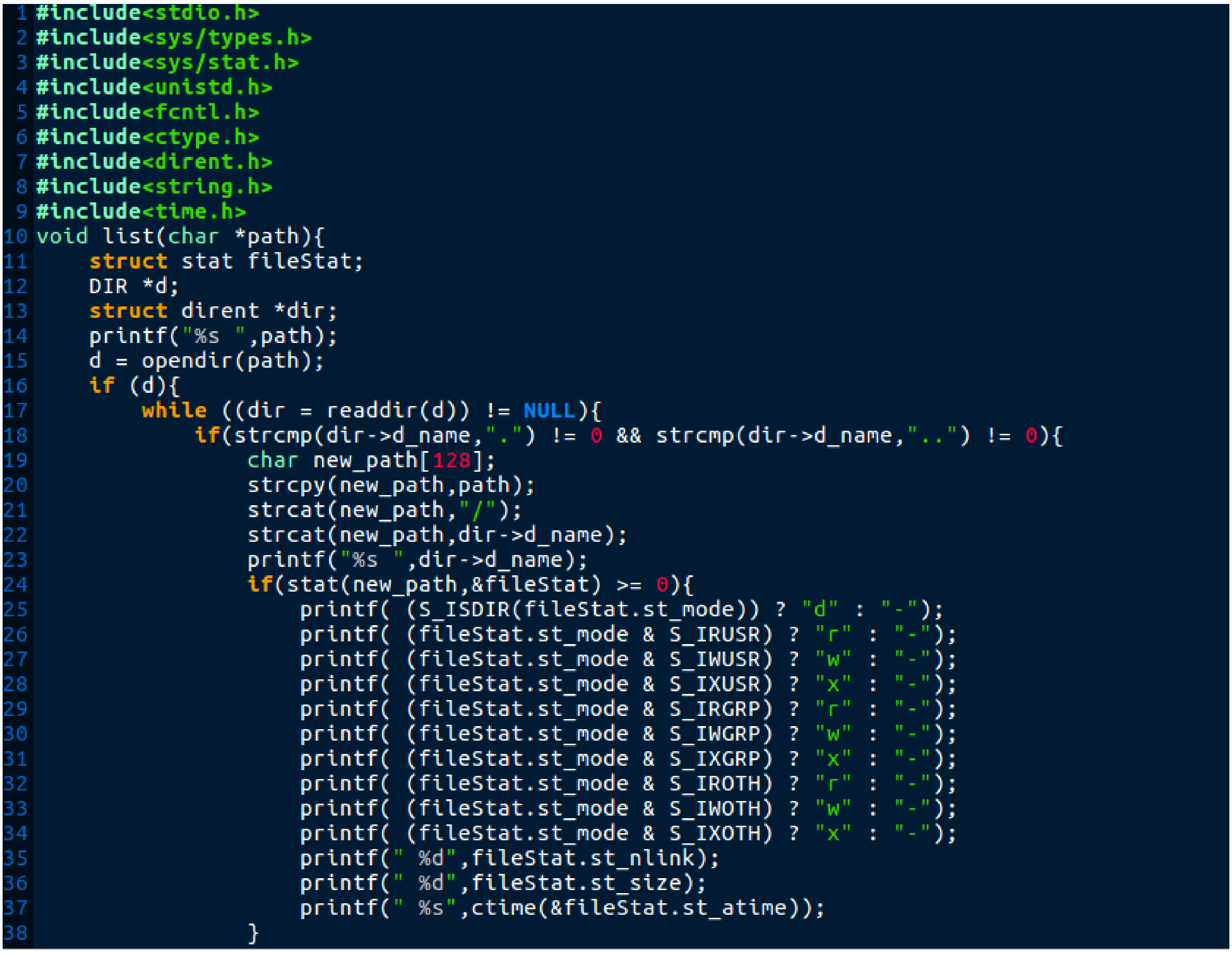
Team Members:
Vijesh Ghandare 111403013
Nikhil Gawande 111408013
Anupam Godse 111408016

Q1.1. “Using APIs, write the equivalent program for the following shell script.

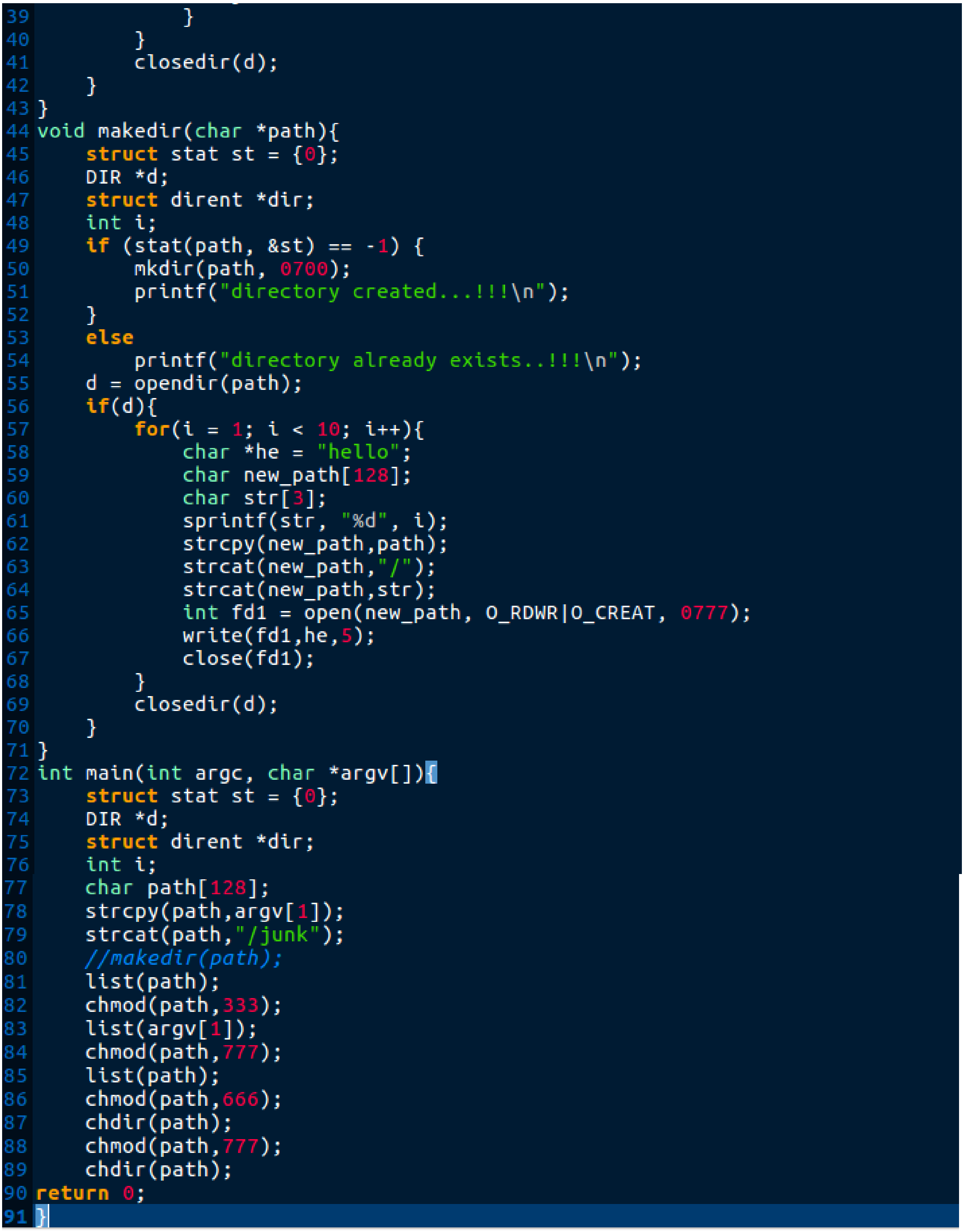
```
mkdir junk
for i in 1 2 3 4 5 do
echo hello >junk/$i
done
ls –l junk
chmod –r junk
ls –l
chmod +r junk
ls –l junk
chmod –x junk
cd junk
chmod +x junk
cd junk
```

Write appropriate comments in the program to observe the execution output.

CODE:

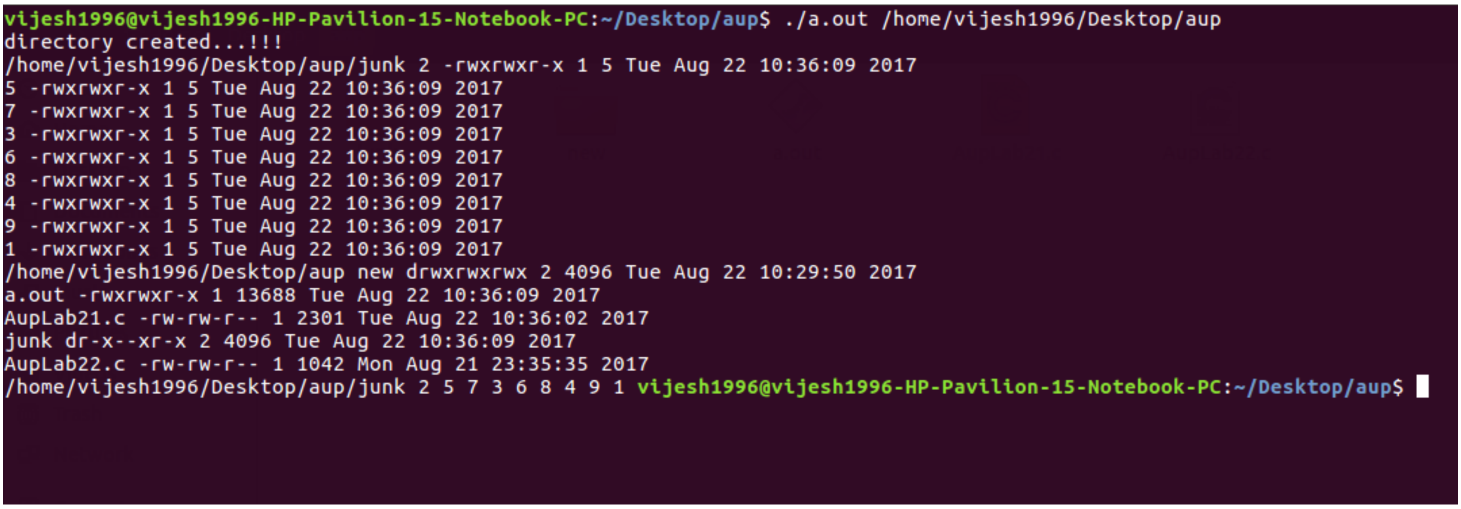


```c
#include<stdio.h>
#include<sys/types.h>
#include<sys/stat.h>
#include<unistd.h>
#include<fcntl.h>
#include<ctype.h>
#include<dirent.h>
#include<string.h>
#include<time.h>
void list(char *path){
    struct stat fileStat;
    DIR *d;
    struct dirent *dir;
    printf("%s ",path);
    d = opendir(path);
    if (d){
        while ((dir = readdir(d)) != NULL){
            if(strcmp(dir->d_name,".") != 0 && strcmp(dir->d_name,"..") != 0){
                char new_path[128];
                strcpy(new_path,path);
                strcat(new_path,"/");
                strcat(new_path,dir->d_name);
                printf("%s ",dir->d_name);
                if(stat(new_path,&fileStat) >= 0){
                    printf( (S_ISDIR(fileStat.st_mode)) ? "d" : "-");
                    printf( (fileStat.st_mode & S_IRUSR) ? "r" : "-");
                    printf( (fileStat.st_mode & S_IWUSR) ? "w" : "-");
                    printf( (fileStat.st_mode & S_IXUSR) ? "x" : "-");
                    printf( (fileStat.st_mode & S_IRGRP) ? "r" : "-");
                    printf( (fileStat.st_mode & S_IWGRP) ? "w" : "-");
                    printf( (fileStat.st_mode & S_IXGRP) ? "x" : "-");
                    printf( (fileStat.st_mode & S_IROTH) ? "r" : "-");
                    printf( (fileStat.st_mode & S_IWOTH) ? "w" : "-");
                    printf( (fileStat.st_mode & S_IXOTH) ? "x" : "-");
                    printf(" %d",fileStat.st_nlink);
                    printf(" %d",fileStat.st_size);
                    printf(" %s",ctime(&fileStat.st_atime));
                }
```



```
            }
        }
        closedir(d);
    }
}
void makedir(char *path){
    struct stat st = {0};
    DIR *d;
    struct dirent *dir;
    int i;
    if (stat(path, &st) == -1) {
        mkdir(path, 0700);
        printf("directory created...!!!\n");
    }
    else
        printf("directory already exists..!!!\n");
    d = opendir(path);
    if(d){
        for(i = 1; i < 10; i++){
            char *he = "hello";
            char new_path[128];
            char str[3];
            sprintf(str, "%d", i);
            strcpy(new_path,path);
            strcat(new_path,"/");
            strcat(new_path,str);
            int fd1 = open(new_path, O_RDWR|O_CREAT, 0777);
            write(fd1,he,5);
            close(fd1);
        }
        closedir(d);
    }
}
int main(int argc, char *argv[]){
    struct stat st = {0};
    DIR *d;
    struct dirent *dir;
    int i;
    char path[128];
    strcpy(path,argv[1]);
    strcat(path,"/junk");
    //makedir(path);
    list(path);
    chmod(path,333);
    list(argv[1]);
    chmod(path,777);
    list(path);
    chmod(path,666);
    chdir(path);
    chmod(path,777);
    chdir(path);
return 0;
}
```

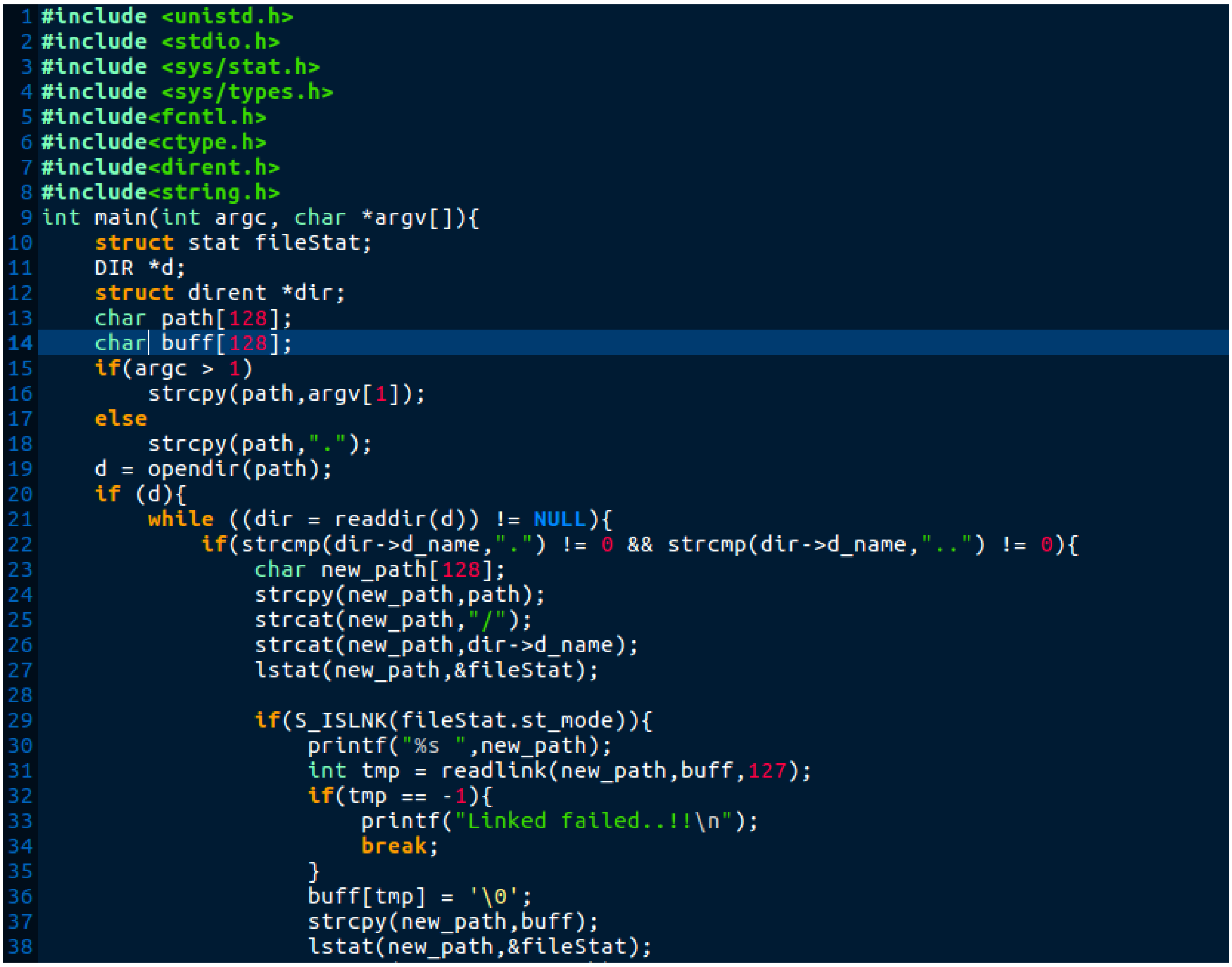
Outputs:



```
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ ./a.out /home/vijesh1996/Desktop/aup
directory created...!!!
/home/vijesh1996/Desktop/aup/junk 2 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
5 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
7 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
3 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
6 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
8 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
4 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
9 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
1 -rwxrwxr-x 1 5 Tue Aug 22 10:36:09 2017
/home/vijesh1996/Desktop/aup new drwxrwxrwx 2 4096 Tue Aug 22 10:29:50 2017
a.out -rwxrwxr-x 1 13688 Tue Aug 22 10:36:09 2017
AupLab21.c -rw-rw-r-- 1 2301 Tue Aug 22 10:36:02 2017
junk dr-x--xr-x 2 4096 Tue Aug 22 10:36:09 2017
AupLab22.c -rw-rw-r-- 1 1042 Mon Aug 21 23:35:35 2017
/home/vijesh1996/Desktop/aup/junk 2 5 7 3 6 8 4 9 1 vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$
```

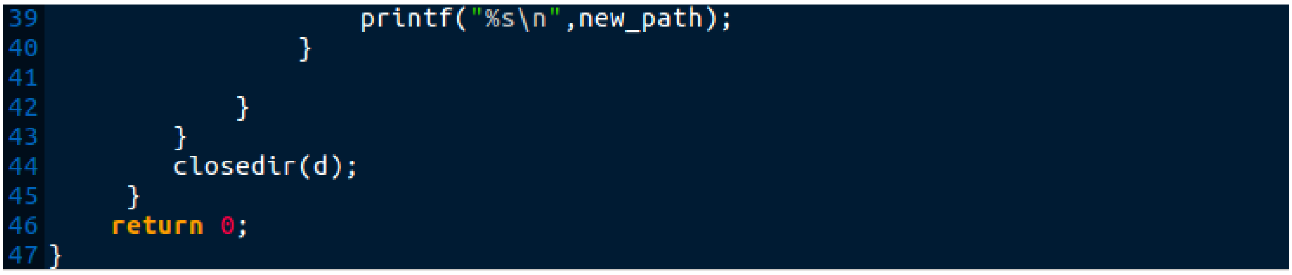
2. A function realpath() resolves all symbolic links in path and returns the ultimate target. Write a program to list the ultimate target of the only filenames that are symbolic links in a directory. The program takes one optional argument, which is the name of a directory to be searched for the links. When no argument is specified, the search is conducted in the current working directory. Display appropriate error messages.

CODE:



```c
#include <unistd.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include<fcntl.h>
#include<ctype.h>
#include<dirent.h>
#include<string.h>
int main(int argc, char *argv[]){
    struct stat fileStat;
    DIR *d;
    struct dirent *dir;
    char path[128];
    char buff[128];
    if(argc > 1)
        strcpy(path,argv[1]);
    else
        strcpy(path,".");
    d = opendir(path);
    if (d){
        while ((dir = readdir(d)) != NULL){
            if(strcmp(dir->d_name,".") != 0 && strcmp(dir->d_name,"..") != 0){
                char new_path[128];
                strcpy(new_path,path);
                strcat(new_path,"/");
                strcat(new_path,dir->d_name);
                lstat(new_path,&fileStat);

                if(S_ISLNK(fileStat.st_mode)){
                    printf("%s ",new_path);
                    int tmp = readlink(new_path,buff,127);
                    if(tmp == -1){
                        printf("Linked failed..!!\n");
                        break;
                    }
                    buff[tmp] = '\0';
                    strcpy(new_path,buff);
                    lstat(new_path,&fileStat);
```

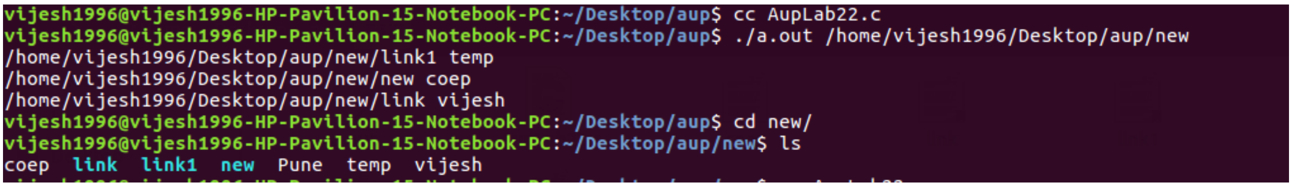


```
                                printf("%s\n",new_path);
                        }

                }
        }
        closedir(d);
    }
    return 0;
}
```

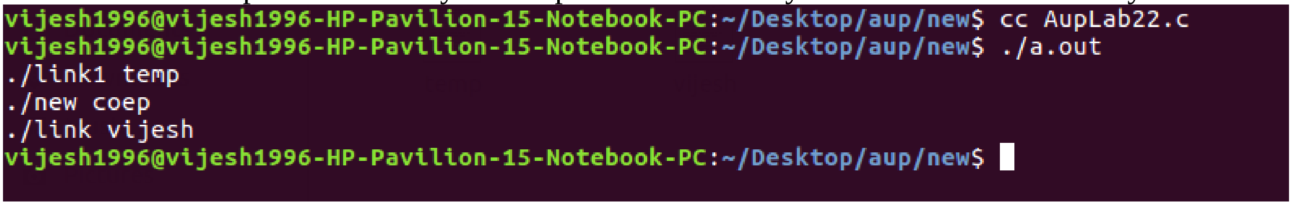
OUTPUT:

Case I: when the path of directory is specified then return the realpath for symlinks



```
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ cc AupLab22.c
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ ./a.out /home/vijesh1996/Desktop/aup/new
/home/vijesh1996/Desktop/aup/new/link1 temp
/home/vijesh1996/Desktop/aup/new/new coep
/home/vijesh1996/Desktop/aup/new/link vijesh
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ cd new/
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup/new$ ls
coep  link  link1  new  Pune  temp  vijesh
```

Case II: When the path of directory is not specified the find symlinks in current directory.

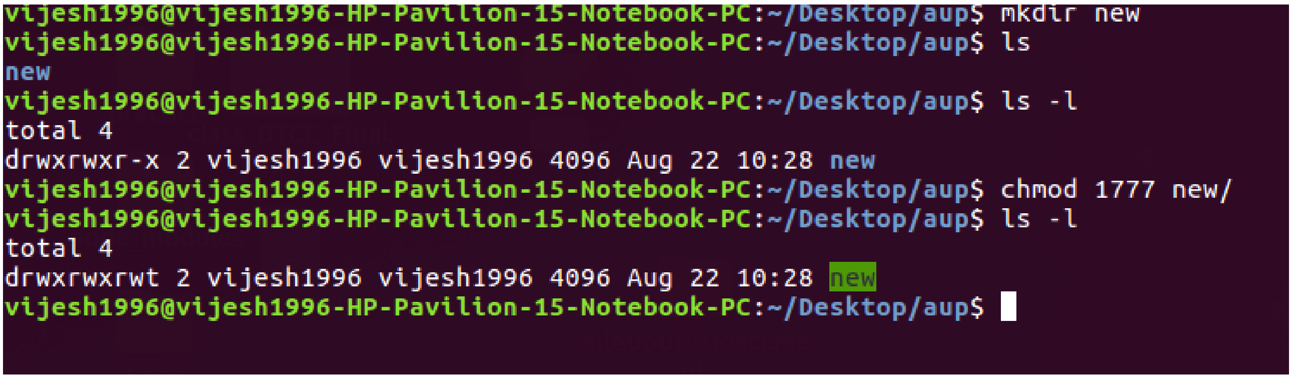


```
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup/new$ cc AupLab22.c
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup/new$ ./a.out
./link1 temp
./new coep
./link vijesh
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup/new$
```

---

3. Create a shared directory for usage with a purpose that any user (not super user) can create new files in this directory, but only the owner can delete his own files and everyone else can read all files?

CODE and Output:



```
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ mkdir new
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ ls
new
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ ls -l
total 4
drwxrwxr-x 2 vijesh1996 vijesh1996 4096 Aug 22 10:28 new
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ chmod 1777 new/
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$ ls -l
total 4
drwxrwxrwt 2 vijesh1996 vijesh1996 4096 Aug 22 10:28 new
vijesh1996@vijesh1996-HP-Pavilion-15-Notebook-PC:~/Desktop/aup$
```