PRACTICAL : 3 & ASSIGNMENT : 2

* **Aim :**

Write a program for ‘ls’ command using ‘opendir()’ and ‘readdir()’ system call. Also Create a program that can be used to implement following specification for “ls” command.

* Ls
* Ls -a
* Ls -A
* Ls -R
* **Code :**

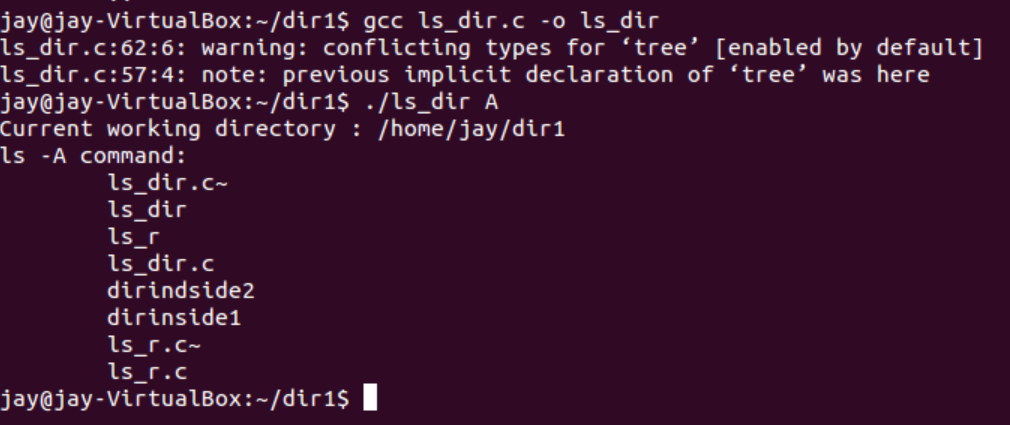
#include<stdio.h>  
#include<stdlib.h>  
#include<dirent.h>  
#include<unistd.h>  
#include<limits.h>  
#include<string.h>  
  
int main(int argc,char\* argv[])  
{  
        struct dirent \*dp,\*dp2;  
        DIR \*dir,\*dir2;  
        char cwd[PATH\_MAX];  
        if(getcwd(cwd,sizeof(cwd))!=NULL)  
        {  
                printf("Current working directory : %s",cwd);  
        }  
  
        if(strcmp(argv[1],"a")==0)  
        {  
  
                dir=opendir(cwd);  
                if(dir==NULL)  
                {  
                        printf("Invalid directory..\n");  
                        exit(1);  
                }  
                else  
                {  
                        printf("\nls -a command:\n");  
                        while((dp=readdir(dir))!=NULL)  
                                printf("\t%s\n",dp->d\_name);  
                }  
                closedir(dir);  
        }  
        else if(strcmp(argv[1],"A")==0)  
        {  
                dir2=opendir(cwd);  
                if(dir2==NULL)  
                {  
                printf("Invalid directory..\n");  
                exit(1);  
                }  
                else  
                {  
                        printf("ls -A command:\n");  
                        while((dp=readdir(dir2))!=NULL)  
                        {  
        if((strcmp(dp->d\_name,"..")!=0)&&(strcmp(dp->d\_name,".")!=0))  
                                printf("\t%s\n",dp->d\_name);  
                        }  
                }  
                closedir(dir2);  
        }  
  
        else if(strcmp(argv[1],"R")==0)  
        {  
                 ls\_dir(cwd, 0);  
        }  
        return 0;  
}  
void ls\_dir(char \*b\_Path, const int root)  
{  
    int i;  
    char path[1024];  
    struct dirent \*dp;  
    DIR \*dir = opendir(b\_Path);  
  
    if (!dir)  
        return;  
    while ((dp = readdir(dir)) != NULL)  
    {  
        if (strcmp(dp->d\_name, ".") != 0 && strcmp(dp->d\_name, "..") != 0)  
        {  
            for (i=0; i<root; i++)  
            {  
                if (i%2 == 0 || i == 0){

                        printf("\t");}  
                else  
                    printf(" ");  
            }  
        printf("\t%s\n", dp->d\_name);  
  
            strcpy(path, b\_Path);  
            strcat(path, "/");  
            strcat(path, dp->d\_name);  
            tree(path, root + 2);  
        }  
    }  
    closedir(dir3);

* **Output :**
* Ls -a command :



* Ls -A command :



* Ls -R command :



* **Conclusion :**
* In this practical we have studied about ls command using opendir() and readdir() system call. Also implemented “ls -a” , “ls -A” , “ls -R ” command.