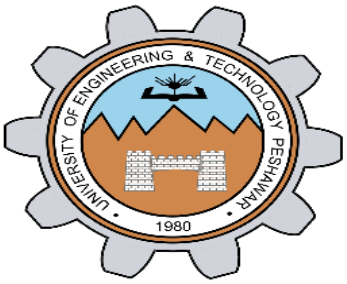
Lab report no 9



Fall 2022

Computer System Programming Lab

Submitted By

Names	**Registration No**
Muhammad Ali	**19pwcse1801**

Section: **A**

Date:19,2,22

Submitted To: MAM. Madiha Sher

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Title: - “Depth-first and breadth-first traverse of directories and files”

Task no 1: -

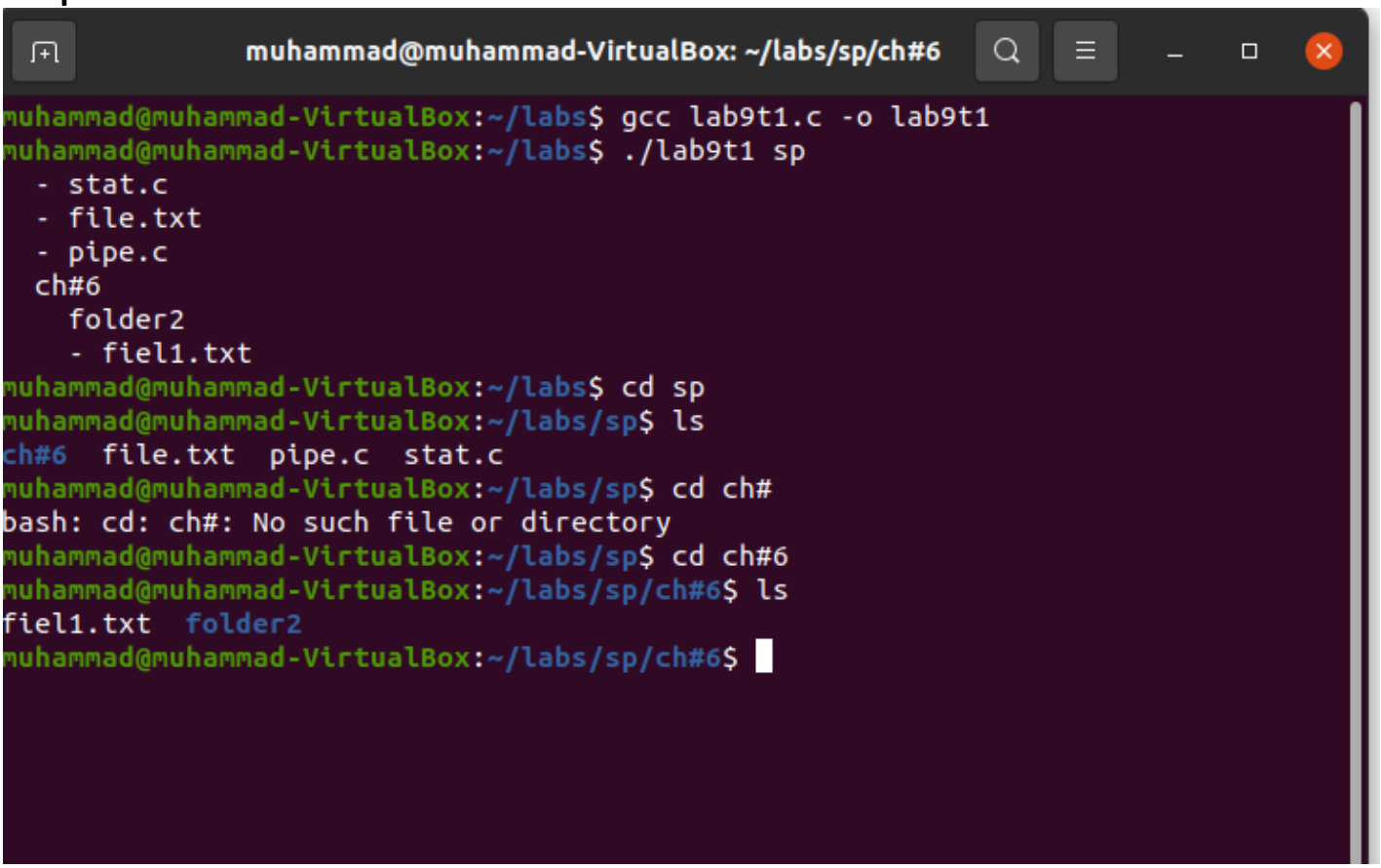
In this task we implement Depth-first traverse of some directories and files from the root a directory as ‘sp’ as shown in the terminal.

Code: -

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

void listdir( char *name, int level)
{
   DIR *dir;
   struct dirent *entry;

   if (!(dir = opendir(name)))
      return;
   if (!(entry = readdir(dir)))
      return;

   do {
      if (entry->d_type == DT_DIR)
       {
         char path[1024];
         int len = snprintf(path, sizeof(path)-1, "%s/%s", name, entry->d_name);
         path[len] = 0;

         if (strcmp(entry->d_name, ".") == 0 || strcmp(entry->d_name, "..") == 0)
            continue;

         printf("%*s%s\n", level*2, "", entry->d_name);
         listdir(path, level + 1);
      }

      else{
         printf("%*s- %s\n", level*2, "", entry->d_name);
      }

   } while (entry = readdir(dir));
   closedir(dir);
```

```
}

 int main(int ar,char *v[] ){
  int a=1;  //levels

  if (ar!=2){
  printf("error: no appropriate arguments...");
  return -1;
  }

  else{
  listdir(v[1],a);
  }
  return 0;

}
```

Output 1: -



Task no 2: -

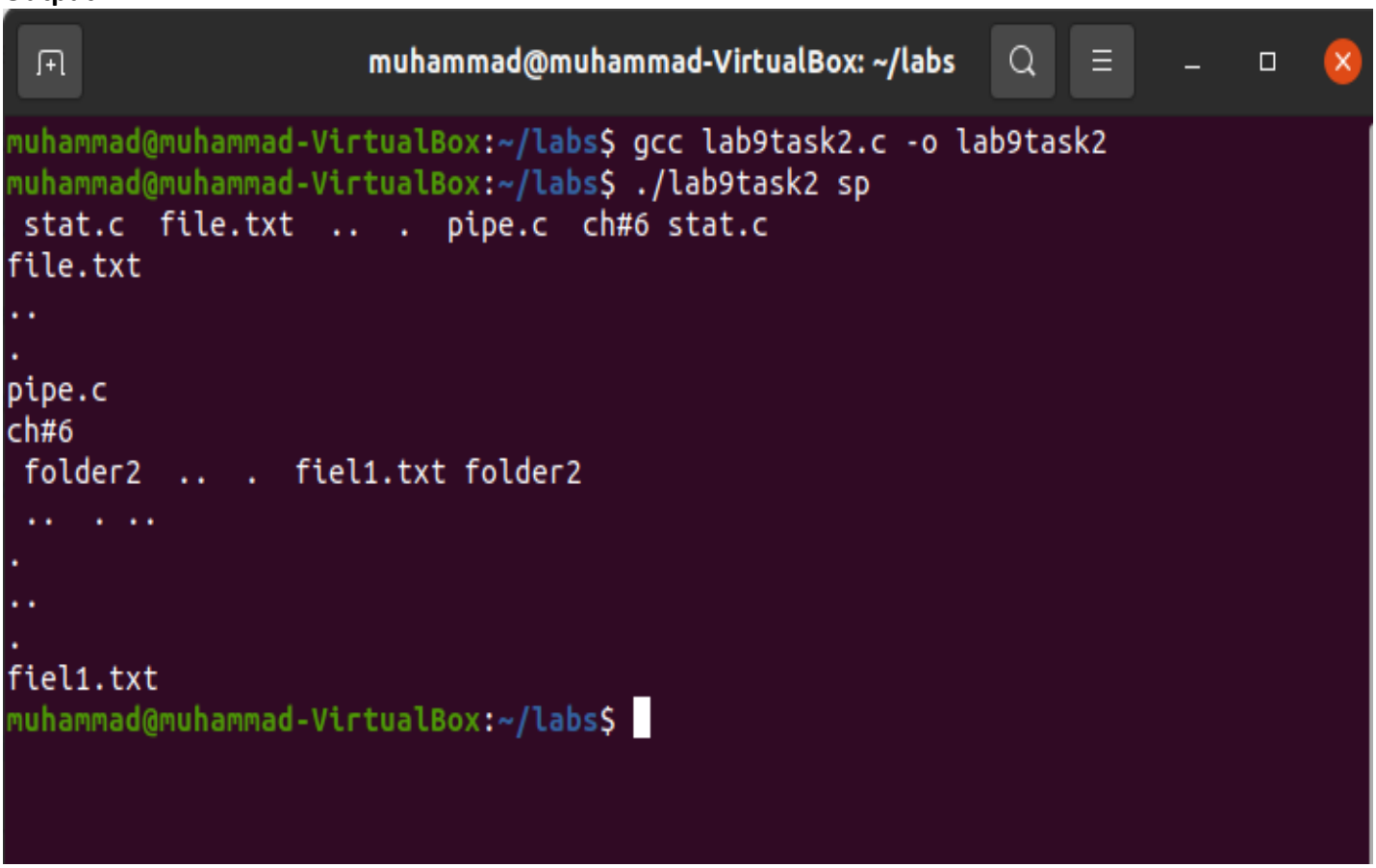
Here we implement breadth-first traversing in the loop which is visited on basis of different levels.

Code: -

```
#include<stdio.h>
#include<unistd.h>
#include<sys/stat.h>
#include<dirent.h>


void breadthfirst(char *dirname){

  struct dirent *ds;
  DIR *dr;
  struct stat buf;

    dr=opendir(dirname);

    if(dr==NULL)
    {
            printf("%s\n",dirname);
            perror("Failed to open dir\n");
    }
    chdir(dirname);

    while ((ds=readdir(dr))!=NULL){

      printf(" %s ",ds->d_name);
}
    rewinddir(dr);
     while ((ds=readdir(dr))!=NULL)
      {
       printf("%s\n",ds->d_name);
       stat (ds->d_name,&buf);

       if(S_ISDIR(buf.st_mode)){
         if (ds->d_name[0]!='.'){
            breadthfirst(ds->d_name);
            chdir("..");
         }
       }
     }
```

```
 }

  int main(int ar, char * v[]){

  if (ar!=2){
 printf("error: no appropriate arguments...");
 return -1;
 }

 else{
 breadthfirst(v[1]);
 }
 return 0;

}
```

Output 2: -



Task no 3: -

In this task we are searching a file during traversing (pfind command implementation).

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

void depthfirst(char *dirname,char * filename){

  struct dirent *ds;
  DIR *dr;
  struct stat buf;

    dr=opendir(dirname);

    if (dr==NULL){
    perror("failed to open: ");
    }
    chdir(dirname);

    while ((ds=readdir(dr))!=NULL){

       stat (ds->d_name,&buf);
       if (!strcmp(ds->d_name,filename)){
        printf("  file '%s' is availible in given directry..\n",filename);
        exit(0);
     }


     if(S_ISDIR(buf.st_mode)){
       if (ds->d_name[0]!='.'){
         depthfirst(ds->d_name,filename);
         chdir("..");

       }
     }
    }
```

```
    closedir(dr);
 }

  int main(int ar, char * v[]){

   if (ar!=3){
    printf("program required two argument...");
    }

    else
    {
     depthfirst(v[1],v[2]);
     }

  depthfirst(v[1],v[2]);
  return 0;

}
```

Output 3: -