LABORATORUL 6

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

#define BSIZE 16384

void afis(char *name)
{


    DIR*d;
    struct dirent *dir;
    d=opendir(name);
    if(d)
    {
        while((dir = readdir(d))!= NULL){

            if((strstr(dir->d_name,".")==0)&& dir->d_namlen>=3)
            {
            char* temp = (char *)malloc(BSIZE*sizeof(char));
            strcpy(temp,name);
            // printf("\ntemp este %s  ",temp);
              strcat(temp,"\\");
             // printf("\ntemp este %s  ",temp);
             if (dir->d_name != "." && dir->d_name != ".."){
              printf("\n%s",dir->d_name);
              afis((strcat(temp,dir->d_name)));
             }

            }
            else
            {

            printf("\n%s",dir->d_name);
                continue;
            }
        }
```

```
        closedir(d);
    }

}


int main(void)
{
       char* mode;
//    FILE fin, *fout; /*Input and output handles */
//    if(argc>=3)
//    {
//        if(strcmp(argv[1],"-lower")==0)
//        {
//         mode="a+";}
//        }
//    char buf[BSIZE];
//    int count;
//    fin = fopen(argv[2], "r");
//    fout = fopen(argv[3], mode);
//
//    if(fin == NULL || fout == NULL)
//    {
//    // if not, exit with error code 1
//    return 1;
//    }
//    // while there are still bytes to be read, copy contents
//    while ((count = fread(buf, 1, BSIZE, fin)) > 0)
//        {
//            int i;
//            for( i=0;i<BSIZE;i++)
//            {
//                if( buf[i]>=65 &&  buf[i]<=90)
//                    buf[i]= buf[i]+32;
//            }
//            fwrite(buf, 1, count, fout);
//        }
//    fclose(fin);
//    fclose(fout);
    char* cwd = NULL;
    cwd=getcwd(cwd,BSIZE);
    printf("%s ->  dir curent \n ",cwd);
    afis(cwd);


    return 0;
```

```
}
```