

System Programming Homework - Project Report

Terminal Based File Manager

Student Information:

Demirhan Yılal - B211202017

Ahmetcan Ersoy - B211202030

Faruk Emre Ekşioğlu - B211202045

Github Link:

<https://github.com/femrek/system-programming-hw>

Project Description:

The development of the **Terminal Based File Manager** involved a systematic approach to achieve the project goals and implement all required functionalities using modular programming. Each module was designed to handle specific tasks, ensuring separation of concerns, maintainability, and extensibility.

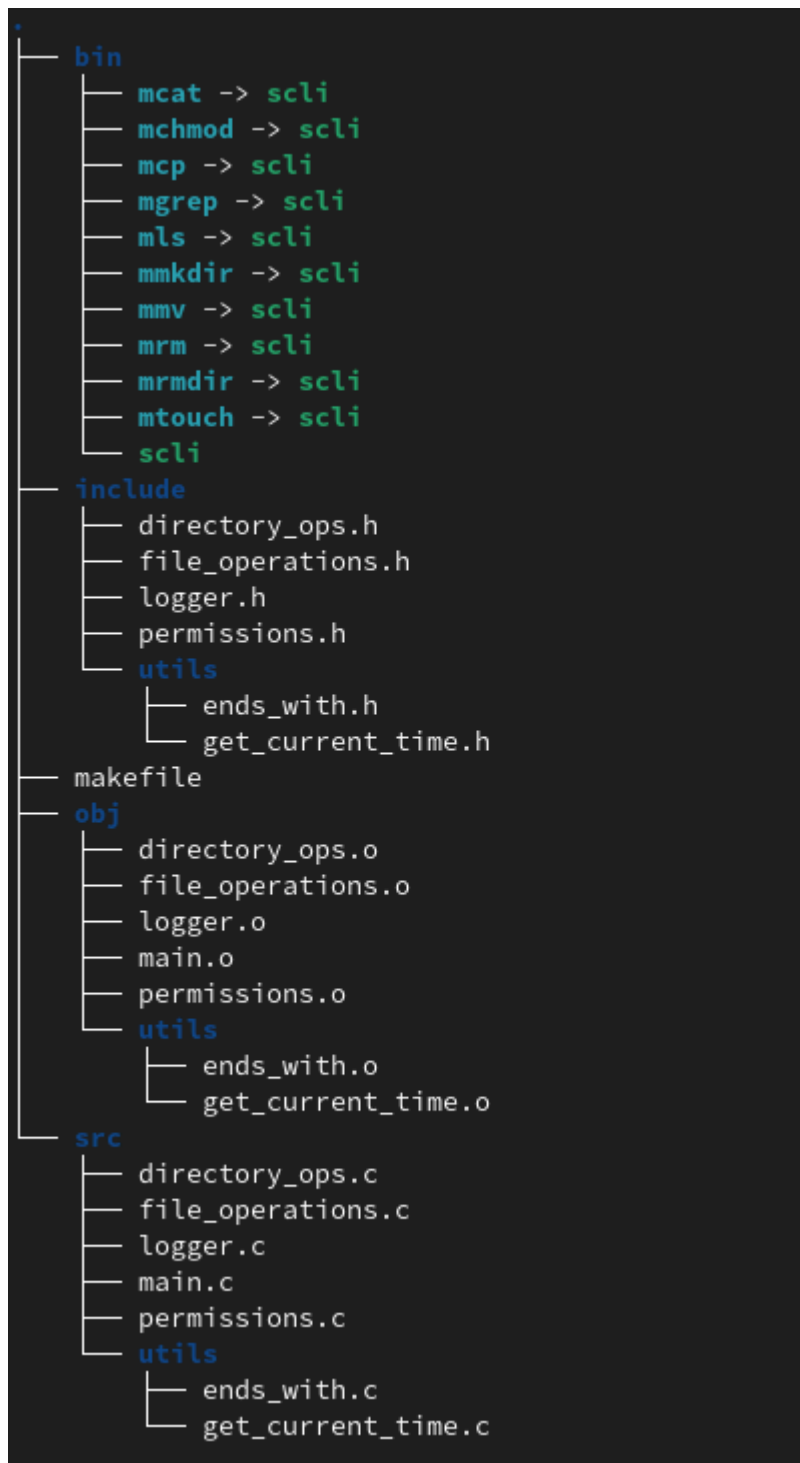


Figure - 1

The project tree shown in Figure - 1. We develop the project modular. In file_operations.c we did touch, rm, cp, mv, cat, and grep operations. In directory_ops.c we did ls, mkdir, rmdir operations. permissions.c includes chmod operation. Also a logger

is provided, and we make information and debug logging operations.

file_operations.h:

```
int file_touch(int argc, char *argv[]);
int file_rm(int argc, char *argv[]);
int file_cp(int argc, char *argv[]);
int file_mv(int argc, char *argv[]);
int file_cat(int argc, char *argv[]);
int file_grep(int argc, char *argv[]);
```

directory_ops.h

```
int dir_ls(int argc, char *argv[]);
int dir_mkdir(int argc, char *argv[]);
int dir_rmdir(int argc, char *argv[]);
```

permissions.h

```
int permission_chmod(int argc, char *argv[]);
```

logger.h

```
void logI(char*, char*);
void logD(char*, char*);
```

Function Explanations:

1- main.c:

In main.c which acts as dispatcher that routes the program's execution based on the name of the executable being invoked, we take argv[0] parameter (**Figure - 4**), which represents the name of the invoked executable or soft link, is compared against each entry in the executables array using the **ends_with (Figure -**

2) function (it is used to check whether the executable name (**argv[0]**) **matches one of the dummy executable** names in the executables array (**Figure - 3**)).

This design allows a single binary to be reused for multiple file management operations by creating symbolic links

```
int ends_with(const char *str, const char *suffix) {  
    size_t str_len = strlen(str);  
    size_t suffix_len = strlen(suffix);  
  
    return (str_len >= suffix_len) &&  
        (!memcmp(str + str_len - suffix_len, suffix, suffix_len));  
}
```

Figure - 2

```
// the list for keep all the executable names. (comes from soft links)  
char *executables[] = {  
    "mls", // 0 -> ls  
    "mkdir", // 1 -> mkdir  
    "rmdir", // 2 -> rmdir  
    "touch", // 3 -> touch  
    "rm", // 4 -> rm  
    "cp", // 5 -> cp  
    "mv", // 6 -> mv  
    "cat", // 7 -> cat  
    "grep", // 8 -> grep  
    "chmod" // 9 -> chmod  
};
```

Figure - 3

```

int main(int argc, char *argv[]) {
    if (argc < 1) {
        perror("No Argument\n");
        return -1;
    }

    // trigger the function according to called soft link file (dummy ex
    char *executable = argv[0];
    logI(executable, NULL);
    if (ends_with(executable, executables[0])) {
        return dir_ls(argc, argv);
    }
    if (ends_with(executable, executables[1])) {
        return dir_mkdir(argc, argv);
    }
    if (ends_with(executable, executables[2])) {
        return dir_rmdir(argc, argv);
    }
    if (ends_with(executable, executables[3])) {
        return file_touch(argc, argv);
    }
    if (ends_with(executable, executables[4])) {
        return file_rm(argc, argv);
    }
    if (ends_with(executable, executables[5])) {
        return file_cp(argc, argv);
    }
    if (ends_with(executable, executables[6])) {
        return file_mv(argc, argv);
    }
    if (ends_with(executable, executables[7])) {
        return file_cat(argc, argv);
    }
    if (ends_with(executable, executables[8])) {
        return file_grep(argc, argv);
    }
    if (ends_with(executable, executables[9])) {
        return permission_chmod(argc, argv);
    }
}

```

Figure - 4

directory_ops.c:

The operations performed are dir_ls, dir_mkdir, dir_rmdir.

1. **dir_ls(int argc, char *argv[])** → opendir(), readdir(), closedir().
 - a. Used for listing files in a given directory.
2. **dir_mkdir(int argc, char *argv[])** → mkdir().
 - a. Used for creating a directory in the desired path.
3. **dir_rmdir(int argc, char *argv[])** → rmdir().
 - a. Used for deleting a given directory.

file_operations.c:

The operations performed are file_touch, file_rm, file_cp, file_mv, file_cat, and file_grep.

1. **file_touch(int argc, char *argv[])** → stat(), creat().
 - a. Used for creating a new file.
2. **file_rm(int argc, char *argv[])** → unlink().
 - a. Used for deleting a file.
3. **file_cp(int argc, char *argv[])** → open(), creat(), read(), write().
 - a. Used for copying a file.
4. **file_mv(int argc, char *argv[])** → rename().
 - a. Used for moving/renaming a file.
5. **file_cat(int argc, char *argv[])** → open(), read(), write().
 - a. Used displaying content of a file.
6. **file_grep(int argc, char *argv[])** → open(), read(), write().
 - a. Used for searching patterns in a given directory.

permissions.c:

The operations performed are permissions_chmod.

1. **permission_chmod(int argc, char *argv[])** → stat(), chmod().
 - a. Used for changing permissions of a file/folder.

logger.c:

The operations performed are logI, logD.

1. **logI(char *op_type, char *text)** → open(), write().
 - a. Used for logging info.
2. **logD(char *op_type, char *text)** → open(), write().
 - a. Used for logging debug.

Outputs:

```
~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:05:46
$ ls
mcat mchmod mcp mgrep mls mmkdir mmv mrm mrmdir mtouch scli

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:06:17
$ ./mkdir ../../deneme
Folder '../../deneme' created successfully.

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:06:39
$ ./mtouch ../../deneme/text.txt
file created successfully.

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:07:26
$ ./mv ../../deneme/text.txt ../../deneme/changed_name_text.txt
file moved successfully.

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:08:59
$ echo "text in text" > ../../deneme/changed_name_text.txt

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:10:28
$ ./mcat ../../deneme/changed_name_text.txt
text in text

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:11:04
$ ./mls ../../deneme
Directory rwxr-xr-x    4096 KB ..
File      rw-r--r--      13 B changed_name_text.txt
Directory rwxr-xr-x    4096 KB .

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:11:16

~/lectures/system-programming/hw/system-programming-hw/bin on 📄 main 15:13:22
$ ./mgrep text ../../deneme/changed_name_text.txt
text in text

~/lectures/system-programming/hw/deneme 15:16:45
$ ls -la
total 12
drwxr-xr-x. 2 demirhan demirhan 4096 Dec 29 15:16 .
drwxr-xr-x. 5 demirhan demirhan 4096 Dec 29 15:06 ..
-rw-r--r--. 1 demirhan demirhan  13 Dec 29 15:16 changed_name_text.txt

~/lectures/system-programming/hw/deneme 15:16:49
$ cd ../system-programming-hw/bin
```



```
~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:19:29
$ ./mfs ../../deneme
Directory rwxr-xr-x    4096 KB ..
Directory rwxr-xr-x    4096 KB .

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:19:37
$ ./rmdir ../../deneme
Directory ' ../../deneme' removed successfully.

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:19:46
$ ./mfs ../../
Directory rwxr-xr-x    4096 KB ..
Directory rwxr-xr-x    4096 KB .
Directory rwxr-xr-x    4096 KB hw-trial
Directory rwxr-xr-x    4096 KB system-programming-hw

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:19:54
$

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:18:36
$ ./mchmod 777 ../../deneme/changed_name_text.txt
permission changed successfully. File: ../../deneme/changed_name_text.txt, new permissions: 777

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:18:42
$ ls -la ../../deneme
total 12
drwxr-xr-x. 2 demirhan demirhan 4096 Dec 29 15:16 .
drwxr-xr-x. 5 demirhan demirhan 4096 Dec 29 15:06 ..
-rwxrwxrwx. 1 demirhan demirhan   13 Dec 29 15:16 changed_name_text.txt

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:18:58
$ ./rm ../../deneme/changed_name_text.txt
file removed successfully.

~/lectures/system-programming/hw/system-programming-hw/bin on 🍃 main 15:19:29
```