

# Hw1 Report

Muhammed Yasir Fidan

March 2021

## 1 Report

### Compile and Run

Firstly I compile my program with `gcc -o myFind hw1.c -std=c99 -Wall -D_BSD_SOURCE -D_POSIX_C_SOURCE` in makefile, so there is no warning when using `-Wall` command. Then I run my program with Valgrind for checking there is memory leak or not and there is no any memory leak.

### Taking arguments

I used `getopt` for getting arguments, also I hold booleans for all arguments so I can check which arguments enter or not. For example if user enter `-f` and `-b` argument only `-f` and `-b` booleans will be true and I use this information for my next instructions. Also there is some error checks in `getopt` for checking user enter correct arguments or not. For instance, when user enter a character for `-l` argument my program catch it, print an error message and exit program with `-1` value. I check this argument is a character or a number by using my `checkNumber` function.

### File Attributes

I create a struct for store file attributes. There is 6 attributes; Target directory, filename, file size, file type, permissions and link number. Then when I traverse files I use `lstat` system call and compare file arguments with traversed files attributes corresponding to the user arguments.

### Signal Handler

I used `sigaction` and write a signal handler function for `ctrl+C` signal. When user send a `ctrl+c` signal I set a flag true and don't exit program immediately because there are a lot of allocated memory and open files. So after setting flag When files closed and dynamic memories freed I inform user about a `ctrl+C` signal arrived and exit program.

## **Traverse directory**

I use opendir and readdir functions to open given directory and traverse its sub directories. My recursive traverseDirectory function open given directory path and recursively traverse its subdirectories.

## **Check files**

While traversing files I use lstat system call to check files attributes with given argument attributes in checkFileMatching function. If Their attributes satisfied I print them on screen like a nice formatted tree.

## **Regular expression**

For regular expression I write my own regular expression function for + operator. This function basically check for given argument reg expression filename and match with the file that I read in given directory or not. Also for case insensitivity I write checkInsensitiveCase function for checking files without case sensitive.

## **Printing output Tree**

Printing output tree was hardest part because I don't wanna rewrite same directories more than one. For this I create a 2D string array and put writed directories path to this array. Before writing a file name to the screen first I check if there is already in array or not. If not I print it and store it in array. if its already writted than I don't write it second time.

## **Some example program run**

User can execute program with giving full path of directory like below

Also, if directory in same folder as code user can just give filename like -w testfile

As you can see there is no leak when execute program with valgrind

```
cse312@ubuntu: ~/Desktop/161044056
cse312@ubuntu:~/Desktop/161044056$ ./myFind -w /home/cse312/Desktop/161044056/testfile -f lost+file
testfile
|--z3
|----lostttttttttttttttfile
|----LOSTttttFile
|----LOstttFILE
|--test2
|----losttfile
|----aaa
|-----losttfile
|----lostfile
cse312@ubuntu:~/Desktop/161044056$
```

Figure 1: Example with full path

```
cse312@ubuntu: ~/Desktop/161044056
cse312@ubuntu:~/Desktop/161044056$ ./myFind -w testfile -f lost+file
testfile
|--z3
|----lostttttttttttttttfile
|----LOSTttttFile
|----LOstttFILE
|--test2
|----losttfile
|----aaa
|-----losttfile
|----lostfile
cse312@ubuntu:~/Desktop/161044056$
```

Figure 2: Example with filename

```
cse312@ubuntu: ~/Desktop/161044056
--25833-- REDIR: 0x40c5f40 (libc.so.6:strnlen) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40c7ef0 (libc.so.6:strncasecmp) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40cd870 (libc.so.6:memchr) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40e08b0 (libc.so.6:wcslen) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40c0f0 (libc.so.6: __GI_strchr) redirected to 0x402ce50 (__GI_strchr)
--25833-- REDIR: 0x40c79c0 (libc.so.6:memset) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x4177210 (libc.so.6: __memset_sse2) redirected to 0x4030820 (memset)
--25833-- REDIR: 0x40c5e80 (libc.so.6: __GI_strlen) redirected to 0x402d400 (__GI_strlen)
--25833-- REDIR: 0x40c5520 (libc.so.6: __GI_strchr) redirected to 0x402cf80 (__GI_strchr)
--25833-- REDIR: 0x40c1830 (libc.so.6:malloc) redirected to 0x402a110 (malloc)
--25833-- REDIR: 0x40c5e30 (libc.so.6:strlen) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40cdfd0 (libc.so.6: __strlen_sse2_bsf) redirected to 0x402d3e0 (strlen)
--25833-- REDIR: 0x40c2150 (libc.so.6:calloc) redirected to 0x402c090 (calloc)
--25833-- REDIR: 0x40c57b0 (libc.so.6:strcpy) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40ce190 (libc.so.6: __strcpy_sse3) redirected to 0x402d4c0 (strcpy)
--25833-- REDIR: 0x40c1de0 (libc.so.6:free) redirected to 0x402b370 (free)
--25833-- REDIR: 0x40c7960 (libc.so.6: __GI_memmove) redirected to 0x4030b00 (__GI_memmove)
--25833-- REDIR: 0x40c56e0 (libc.so.6:strcmp) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x4184900 (libc.so.6: __strcmp_sse3) redirected to 0x402e5b0 (strcmp)
--25833-- REDIR: 0x40c52c0 (libc.so.6:strcat) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40d9750 (libc.so.6: __strcat_sse3) redirected to 0x402d080 (strcat)
--25833-- REDIR: 0x40c09a0 (libc.so.6:rindex) redirected to 0x4024580 (_vgnU_ifunc_wrapper)
--25833-- REDIR: 0x40df1d0 (libc.so.6: __strchr_sse2_bsf) redirected to 0x402ce20 (rindex)
testfile
|--z3
----lostttttttttttttfile
----LOSTttttFile
----LOsttttFILE
--test2
----lostttfile
----aaa
----lostttfile
--lostfile
==25833==
==25833== HEAP SUMMARY:
==25833==   in use at exit: 0 bytes in 0 blocks
==25833== total heap usage: 59 allocs, 59 frees, 241,728 bytes allocated
==25833==
==25833== All heap blocks were freed -- no leaks are possible
==25833==
==25833== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
==25833== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
cse312@ubuntu:~/Desktop/161044056$
```

Figure 3: Example with Valgrind