

Programming Exercise – 4

Siri N Shetty – PES2UG22CS556 (Semester-4 Section-J)

Q3) Write a C program to truncate the files in a directory created after a certain date to half its original size. Inputs to the program: directory and date as run time arguments.

Code:

```
1  #define _XOPEN_SOURCE 700
2  #include <stdio.h>
3  #include <stdlib.h>
4  #include <dirent.h>
5  #include <sys/stat.h>
6  #include <unistd.h>
7  #include <string.h>
8  #include <time.h>
9
10 time_t parse_date(char *date_str) {
11     struct tm tm = {0};
12     if (strptime(date_str, "%Y-%m-%d", &tm) == NULL) {
13         fprintf(stderr, "Error parsing date.\n");
14         return -1;
15     }
16     tm.tm_isdst = -1;
17     return mktime(&tm);
18 }
```

```

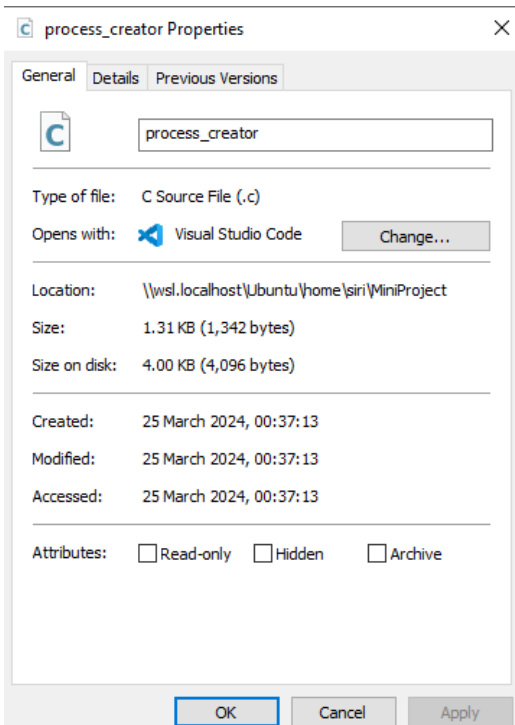
1 void truncate_files(const char *path, time_t specified_time) {
2     DIR *dir = opendir(path);
3     if (!dir) {
4         perror("Failed to open directory");
5         return;
6     }
7
8     struct dirent *entry;
9     struct stat info;
10    char full_path[1024];
11
12    while ((entry = readdir(dir)) != NULL) {
13        if (strcmp(entry->d_name, ".") == 0 || strcmp(entry->d_name, "..") == 0)
14            continue;
15
16        snprintf(full_path, sizeof(full_path), "%s/%s", path, entry->d_name);
17        if (lstat(full_path, &info) == -1) {
18            perror("Failed to get file info");
19            continue;
20        }
21
22        if ($_ISREG(info.st_mode) && info.st_mtime > specified_time) {
23            off_t new_size = info.st_size / 2;
24            // Truncate the file to half of its original size
25            if (truncate(full_path, new_size) == -1) {
26                perror("Failed to truncate file");
27            } else {
28                printf("Truncated '%s' to %ld bytes.\n", full_path, new_size);
29            }
30        }
31    }
32
33    closedir(dir);
34 }
35
36 int main(int argc, char *argv[]) {
37     if (argc != 3) {
38         fprintf(stderr, "Usage: %s <directory> <YYYY-MM-DD>\n", argv[0]);
39         return EXIT_FAILURE;
40     }
41
42     char *directory = argv[1];
43     char *date_str = argv[2];
44     time_t specified_time = parse_date(date_str);
45
46     if (specified_time == -1) {
47         return EXIT_FAILURE;
48     }
49
50     truncate_files(directory, specified_time);
51     return EXIT_SUCCESS;
52 }

```

Output:

```
siri@DESKTOP-F9UMPJU:~/L4$ gcc -D_XOPEN_SOURCE=700 -o truncate_files truncate_files.c
siri@DESKTOP-F9UMPJU:~/L4$ ./truncate_files /home/siri/MiniProject 2024-03-25
Truncated '/home/siri/MiniProject/process_creator.c' to 671 bytes.
Truncated '/home/siri/MiniProject/run_project.sh' to 428 bytes.
Truncated '/home/siri/MiniProject/makefile' to 142 bytes.
Truncated '/home/siri/MiniProject/process_tree_module.c' to 1043 bytes.
siri@DESKTOP-F9UMPJU:~/L4$
```

Before Truncating:



After Truncating:

