CS 283 : Systems Programming
Lab2: File I/O Programming

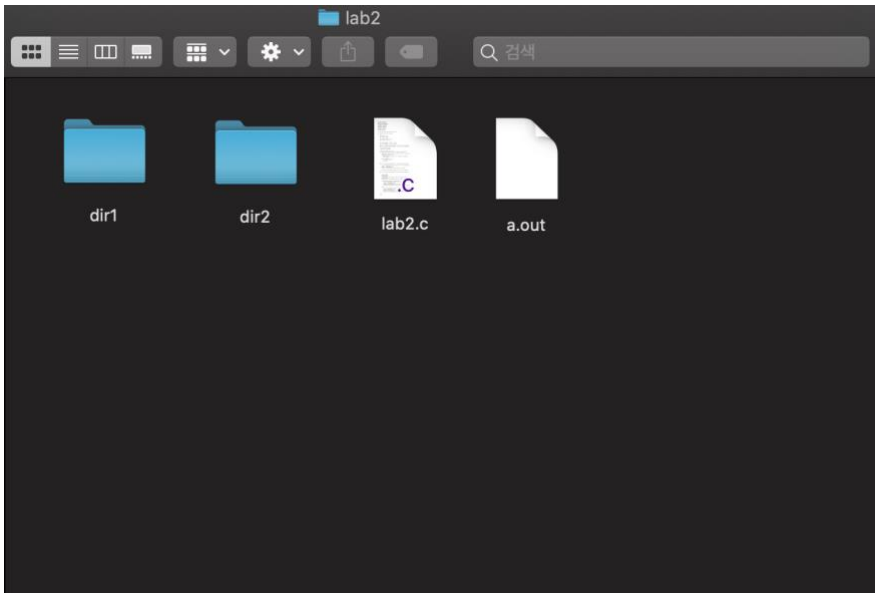
yl3385
Yena Lee

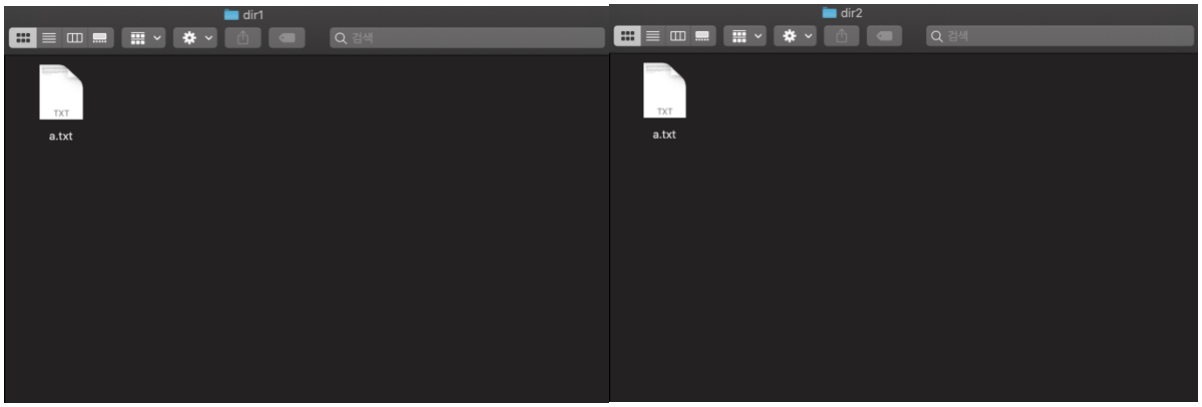
In this assignment, you will write a program to the following specifications, using C file system calls.

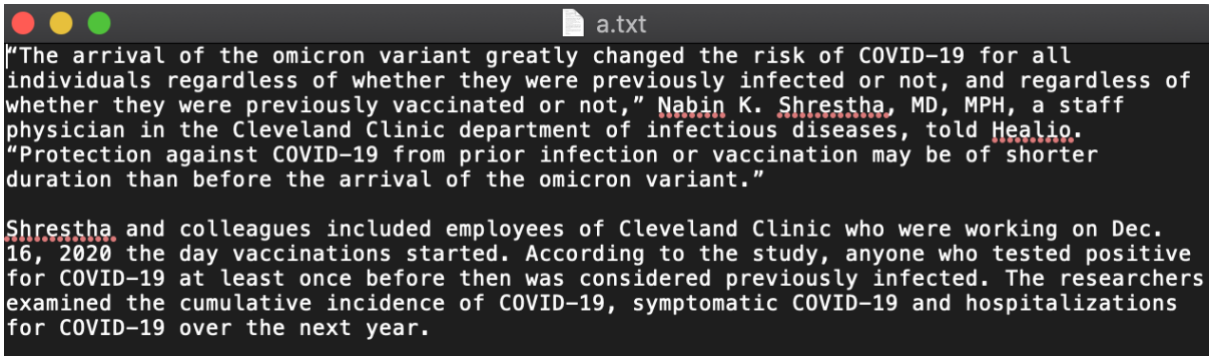
Description

- Given two directories ./a and ./b, passed as parameters to your program, you need to synchronize them, as follows:
    - If a file in a does not exist in directory b, you should replicate it in directory b.
    - If a file in b does not exist in directory a, it should be deleted from directory b.
    - If a file exists in both directories a and b, the file with the most recent modified date / time should be copied from one directory to the other.
- Print a log of your program's activities to stdout or stderr.

<Programming Environment>








```
(base) n3-22-73:lab2 yenalee$ ls
a.out   dir1    dir2    lab2.c
(base) n3-22-73:lab2 yenalee$ cd dir1
(base) n3-22-73:dir1 yenalee$ ls
a.txt
(base) n3-22-73:dir1 yenalee$ cat a.txt
“The arrival of the omicron variant greatly changed the risk of COVID-19 for all individuals
 regardless of whether they were previously infected or not, and regardless of whether they
were previously vaccinated or not,” Nabin K. Shrestha, MD, MPH, a staff physician in the Cle
veland Clinic department of infectious diseases, told Healio. “Protection against COVID-19 f
rom prior infection or vaccination may be of shorter duration than before the arrival of the
 omicron variant.”

Shrestha and colleagues included employees of Cleveland Clinic who were working on Dec. 16,
2020 the day vaccinations started. According to the study, anyone who tested positive for CO
VID-19 at least once before then was considered previously infected. The researchers examine
d the cumulative incidence of COVID-19, symptomatic COVID-19 and hospitalizations for COVID-
19 over the next year.(base) n3-22-73:dir1(base) n3-22-73:dir1 yena(base) n3-22-73:dir1(base
```

<Test Result>

Case #1

If a file in a does not exist in directory b, you should replicate it in directory b.

(a.txt is only in dir1, but not in dir2)

```
(base) n3-22-73:dir1 yenalee$ cd ..
(base) n3-22-73:lab2 yenalee$ ls
a.out   dir1    dir2    lab2.c
(base) n3-22-73:lab2 yenalee$ gcc lab2.c
(base) n3-22-73:lab2 yenalee$ ./a.out
File does not exist in directory 'dir2/a.txt'
File replicated in directory 'dir2/a.txt'
(base) n3-22-73:lab2 yenalee$ cd dir2
(base) n3-22-73:dir2 yenalee$ ls
a.txt
(base) n3-22-73:dir2 yenalee$ cat a.txt
“The arrival of the omicron variant greatly changed the risk of COVID-19 for all individuals
 regardless of whether they were previously infected or not, and regardless of whether they
were previously vaccinated or not,” Nabin K. Shrestha, MD, MPH, a staff physician in the Cle
veland Clinic department of infectious diseases, told Healio. “Protection against COVID-19 f
rom prior infection or vaccination may be of shorter duration than before the arrival of the
 omicron variant.”

Shrestha and colleagues included employees of Cleveland Clinic who were working on Dec. 16,
2020 the day vaccinations started. According to the study, anyone who tested positive for CO
VID-19 at least once before then was considered previously infected. The researchers examine
d the cumulative incidence of COVID-19, symptomatic COVID-19 and hospitalizations for COVID-
19 over the next year.(base) n3-22-73:dir2 yenalee$ cd ..
```

Case #2

If a file in b does not exist in directory a, it should be deleted from directory b

(a.txt is only in dir2, but not in dir1)

```
(base) n3-22-73:lab2 yenalee$ ls
a.out   dir1    dir2    lab2.c
(base) n3-22-73:lab2 yenalee$ gcc lab2.c
(base) n3-22-73:lab2 yenalee$ ./a.out
File does not exist in directory 'dir1/a.txt'
File in directory 'dir2/a.txt' deleted
```

Case #3

If a file exists in both directories a and b, the file with the most recent modified date / time should be copied from one directory to the other.

(a.txt exits in both dir1 and dir2, dir1/a.txt is the most recent modified file.)

dir1/a.txt

a.txt

```
“The arrival of the omicron variant greatly changed the risk of COVID-19 for all
individuals regardless of whether they were previously infected or not, and regardless of
whether they were previously vaccinated or not,” Nabin K. Shrestha, MD, MPH, a staff
physician in the Cleveland Clinic department of infectious diseases, told Healio.
“Protection against COVID-19 from prior infection or vaccination may be of shorter
duration than before the arrival of the omicron variant.”

Shrestha and colleagues included employees of Cleveland Clinic who were working on Dec.
16, 2020 the day vaccinations started. According to the study, anyone who tested positive
for COVID-19 at least once before then was considered previously infected. The researchers
examined the cumulative incidence of COVID-19, symptomatic COVID-19 and hospitalizations
for COVID-19 over the next year.

This is new modified version.
```

```
(base) n3-22-73:lab2 yenalee$ gcc lab2.c
(base) n3-22-73:lab2 yenalee$ ./a.out
dir1's File modified time Sat Jan 29 17:50:20 2022
dir2's File modified time Sat Jan 29 17:46:47 2022
File replicated in directory 'dir2/a.txt'
```

```
(base) n3-22-73:lab2 yenalee$ cd dir2
(base) n3-22-73:dir2 yenalee$ cat a.txt
“The arrival of the omicron variant greatly changed the risk of COVID-19 for all individuals
 regardless of whether they were previously infected or not, and regardless of whether they
were previously vaccinated or not,” Nabin K. Shrestha, MD, MPH, a staff physician in the Cle
veland Clinic department of infectious diseases, told Healio. “Protection against COVID-19 f
rom prior infection or vaccination may be of shorter duration than before the arrival of the
 omicron variant.”

Shrestha and colleagues included employees of Cleveland Clinic who were working on Dec. 16,
2020 the day vaccinations started. According to the study, anyone who tested positive for CO
VID-19 at least once before then was considered previously infected. The researchers examine
d the cumulative incidence of COVID-19, symptomatic COVID-19 and hospitalizations for COVID-
19 over the next year.

This is new modified version.(base) n3-22-73:dir2 yenalee$
```

<Code Description>

lab2.c

```c
#include <stdio.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <stdlib.h>
#include <limits.h>
#include <string.h>
#include <time.h>
#include <fcntl.h>
#include <ftw.h>

int isFileExistsStats(const char *path);

int main(int argn, char *argv[])
{
    char ch;
    char *path1 = NULL;
    char *path2 = NULL;

    char filename[10] = "a.txt";
    FILE *source, *target;


    char dir1_path[20] = "dir1/"; //dir a
    char dir2_path[20] = "dir2/"; //dir b

    path1 = strcat(dir2_path,filename); //a file in dir b dir2/a.txt
    path2 = strcat(dir1_path,filename); //a file in dir a dir1/a.txt

    // printf("%s\n",path1);
    // printf("%s\n",path2);

    //Check if file exists or not
    if (isFileExistsStats(path1) && !isFileExistsStats(path2))
    {
        printf("File does not exist in directory '%s'\n", dir1_path);
        int nResult = remove(path1);
        if(nResult == 0) {
            printf("File in directory '%s' deleted\n",dir2_path);
            path1 = NULL;
        }
        else if (nResult == -1) {
            printf("Fail\n");
            exit(1);
        }
    }
```

First, this program gets file name, and directory path. I made a file named “a.txt”, and put some contents in it. I used directory “dir1, and dir2”, to test. So, this path and file name is concatenated into one path which is (path1, path2).

After that, the program checks the condition if the file exists in specific directory, by the function “isFileExistsStatus”.

```c
/**
 * Function to check whether a file exists or not using
**/
int isFileExistsStats(const char *path)
{
    struct stat stats;
    int exist = stat(path, &stats);

    // Check for file existence
    if (exist == 0)
        return 1;
    else
        return 0;
}
```

If the file exists in the path, it returns 1, and if not it returns 0.

First condition is to check if the file exists in path1, but not in path2. This means a.txt file is in dir2, but not in dir1. If it satisfies the condition, a.txt in dir2 is removed and print “File in directory ‘dir2/a.txt’ delted”.

```
47      else if (isFileExistsStats(path2) && !isFileExistsStats(path1))
48      {
49          printf("File does not exist in directory '%s'\n", dir2_path);
50          source = fopen(path2,"r");
51          target = fopen(path1,"w");
52          while((ch = fgetc(source))!=EOF) fputc(ch, target);
53
54          printf("File replicated in directory '%s'\n", dir2_path);
55      }
56      else if (isFileExistsStats(path2) && isFileExistsStats(path1))
57      {
58
59          struct stat b;
60          struct stat c;
61          stat(path2,&b);
62          printf("dir1's File modified time %s",ctime(&b.st_mtime));
63          stat(path1,&c);
64          printf("dir2's File modified time %s",ctime(&c.st_mtime));
65
66          if(difftime(b.st_mtime,c.st_mtime)>0) {
67              // printf("difftime : %f\n",difftime(b.st_mtime,c.st_mtime));
68              source = fopen(path2,"r");
69              target = fopen(path1,"w");
70              while((ch = fgetc(source))!=EOF) fputc(ch, target);
71              printf("File replicated in directory '%s'\n", dir2_path);
72          }
73          else {
74              // printf("difftime : %f\n",difftime(b.st_mtime,c.st_mtime));
75              source = fopen(path1,"r");
76              target = fopen(path2,"w");
77              while((ch = fgetc(source)) !=EOF) fputc(ch, target);
78              printf("File replicated in directory '%s'\n",dir1_path);
79          }
80      }
81      else {
82
83        printf("File does not exit in anywhere...\n");
84        exit(EXIT_FAILURE);
85
86      }
87      fclose(source);
88      fclose(target);
89      return 0;
90  }
91
```

Second condition is if the file exists in path2 but not in path1. This means a.txt file is in dir1, but not in dir2. In this case, dir1/a.txt replicated in dir2 directory, so dir2 can get same a.txt like dir1 does.

Third condition is the file exits in both path2 and path1. In this case, we should check the last modified date/time in both directory. So, I used st_mtime to calculate last modified time.

stat() stats the file pointed to by *path* and fills in *buf*.

```
struct stat {
    dev_t     st_dev;     /* ID of device containing file */
    ino_t     st_ino;     /* inode number */
    mode_t    st_mode;    /* protection */
    nlink_t   st_nlink;   /* number of hard links */
    uid_t     st_uid;     /* user ID of owner */
    gid_t     st_gid;     /* group ID of owner */
    dev_t     st_rdev;    /* device ID (if special file) */
    off_t     st_size;    /* total size, in bytes */
    blksize_t st_blksize; /* blocksize for file system I/O */
    blkcnt_t  st_blocks;  /* number of 512B blocks allocated */
    time_t    st_atime;   /* time of last access */
    time_t    st_mtime;   /* time of last modification */
    time_t    st_ctime;   /* time of last status change */
};
```

-from linux manual

I also used ctime to change the time in string form.

The call **ctime(*t*)** converts the calendar time *t* into a null-terminated string of the form "Wed Jun 30 21:49:08 1993\n".

To check which one in dir1/a.txt or dir2/a.txt is the recent modified date, I used difftime function .
difftime() is a function that calculates time difference.

```
#include <time.h>

        double difftime(time_t time1, time_t time0);
```

The **difftime**() function returns the number of seconds elapsed between time *time1* and time *time0*, represented as a *double*. Each of the times is specified in calendar time, which means its value is a measurement (in seconds) relative to the Epoch, 1970-01-01 00:00:00 +0000 (UTC).

- from linux manual page

If difftime(path1,path2)>0, I could figure out that path1 time > path2 time, so that path1 time is the last modified time. If dir1/a.txt is the last modified file, the program replicates dir1/a.txt to dir2/a.txt, and if dir2/a.txt is the last modified one, the program copies dir2/a.txt to dir1/a.txt.