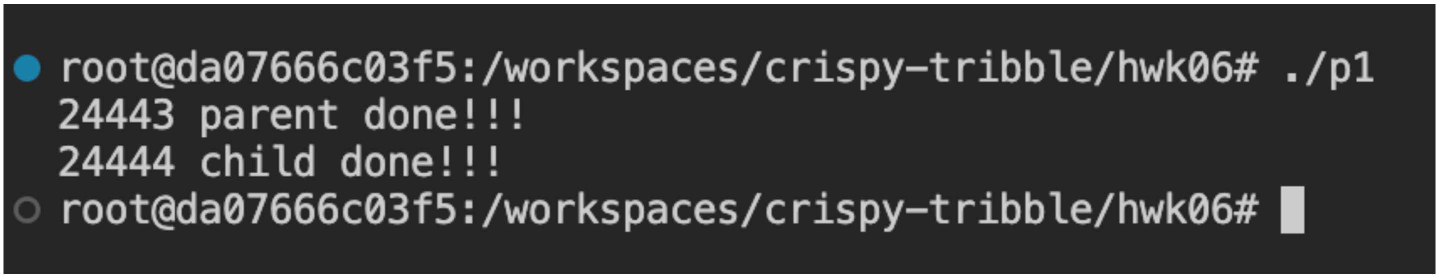
Problem 01

a. Test whether both the child and parent can access the same physical file using the descriptor returned by open(). Explain why (given that the parent and child each has their own stack and data segments)

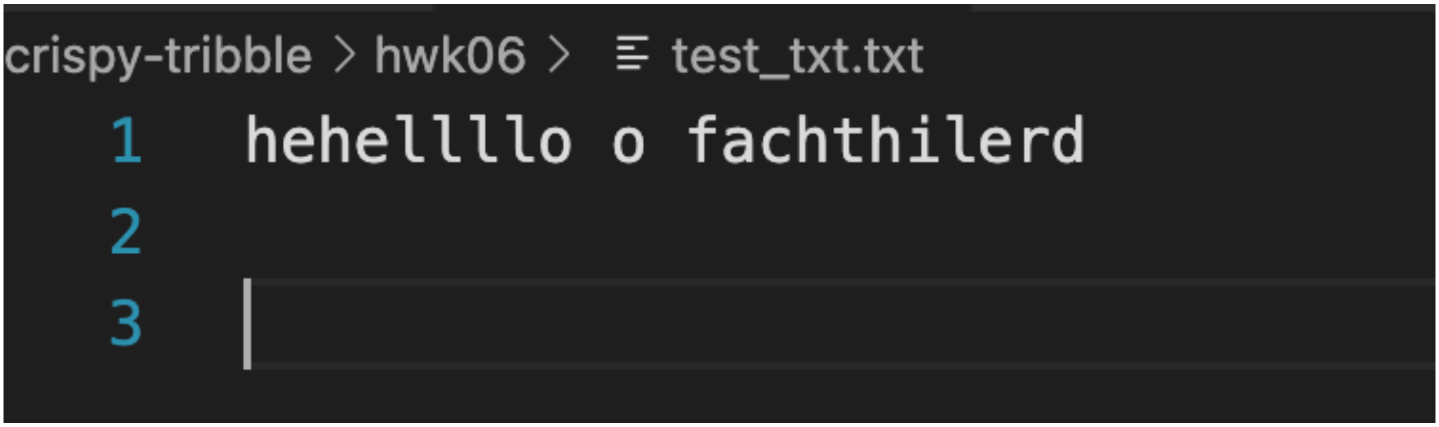
When a file is opened by a process, the operating system creates an entry in the file table, which contains information about the file, such as the file position and mode. Both parent and child process can get the file descriptor from open(); the fd(file descriptor) received from **open()** will be stored in the stack of both children and parent; since each process have their own stack, so they can read and write the fd at the same time.

In conclusion, child process is created by forking the parent process, the child process inherits a copy of the file table entries of the parent process, including the file descriptor returned by the **open()** call. As the below screenshot showed that both process are done, and both of them wrote their message into the file successfully.

However, there are still risks listed below in the answer part of **question b**.



```
● root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p1
  24443 parent done!!!
  24444 child done!!!
○ root@da07666c03f5:/workspaces/crispy-tribble/hwk06# 
```



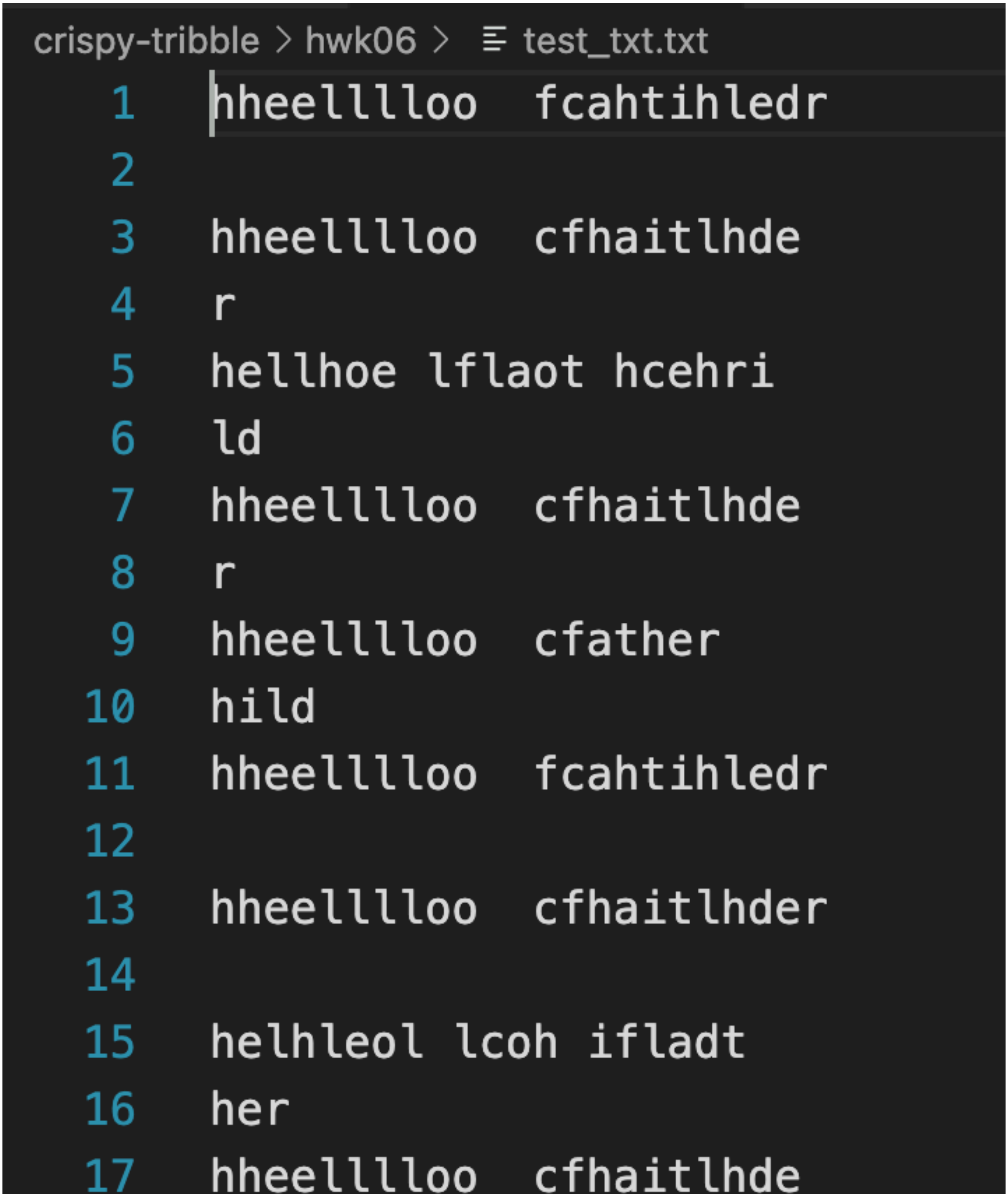
```
hehellllo o fachthilerd


```

b. What happens when both the parent and the child are writing to the file concurrently?

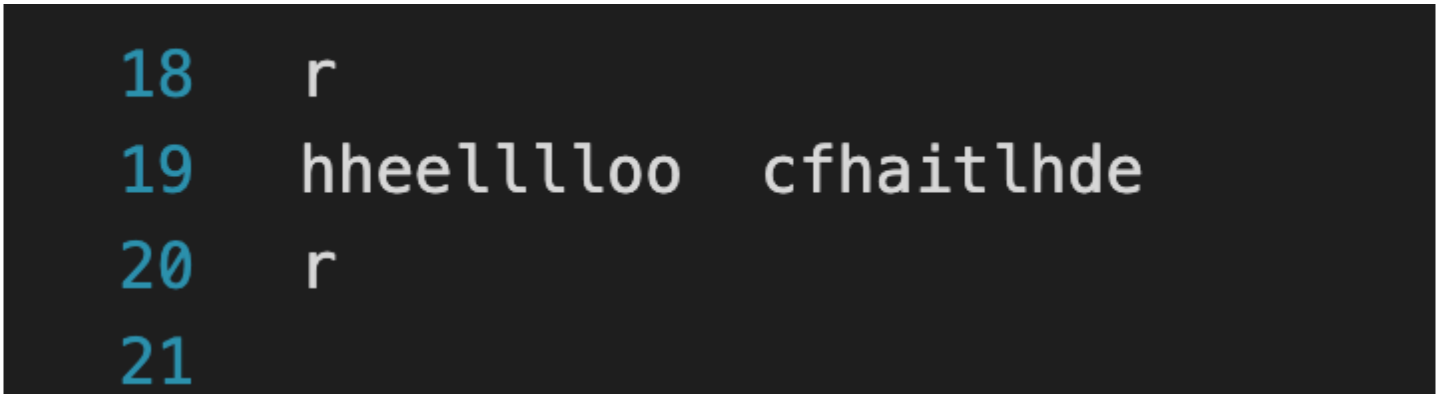
1. The data may be **interleaved**: The data from both processes may be mixed together in an unpredictable way, resulting in corrupted or garbled data.
2. The data may be **overwritten**: If both processes write to the same location within the file, the data from one process may overwrite the data from the other process, resulting in lost or incomplete data. *(not seen in the test program)*

3. The file may become **inconsistent**: If both processes update different parts of the file, the resulting file may not make sense or may be inconsistent. (very common drawbacks)

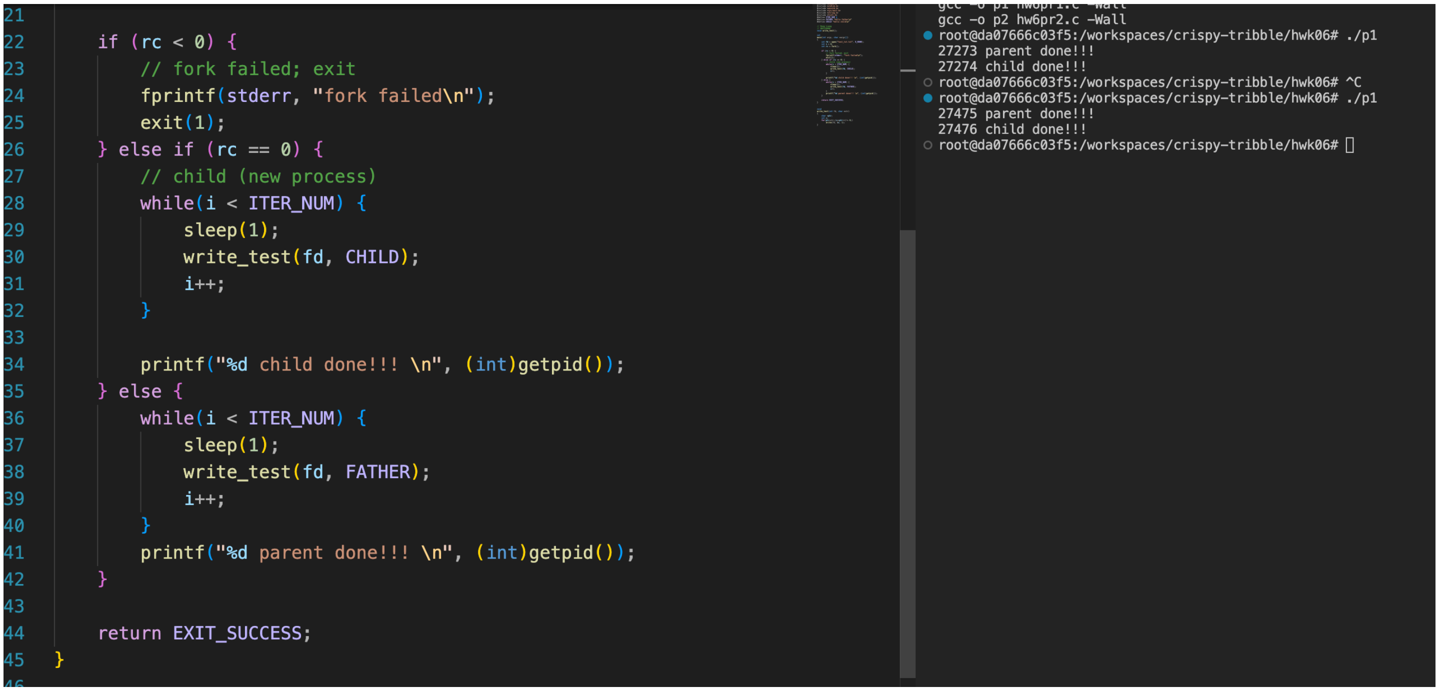
when set the **ITER_NUM** to 10, the file shows like:



```
hheelllloo  fcahtihledr

hheelllloo  cfhaitlhde
r
hellhoe lflaot hcehri
ld
hheelllloo  cfhaitlhde
r
hheelllloo  cfather
hild
hheelllloo  fcahtihledr

hheelllloo  cfhaitlhder

helhleol lcoh ifladt
her
hheelllloo  cfhaitlhde
```



```
r
hheellloo cfhaitlhde
r

```



```
    if (rc < 0) {
        // fork failed; exit
        fprintf(stderr, "fork failed\n");
        exit(1);
    } else if (rc == 0) {
        // child (new process)
        while(i < ITER_NUM) {
            sleep(1);
            write_test(fd, CHILD);
            i++;
        }

        printf("%d child done!!! \n", (int)getpid());
    } else {
        while(i < ITER_NUM) {
            sleep(1);
            write_test(fd, FATHER);
            i++;
        }
        printf("%d parent done!!! \n", (int)getpid());
    }

    return EXIT_SUCCESS;
}
```

```
  gcc -o p1 hw6pr1.c -Wall
  gcc -o p2 hw6pr2.c -Wall
● root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p1
  27273 parent done!!!
  27274 child done!!!
○ root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ^C
● root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p1
  27475 parent done!!!
  27476 child done!!!
○ root@da07666c03f5:/workspaces/crispy-tribble/hwk06# □
```

According to the screen capture: we see the loop enhanced version is working correctly, and we did see the inconsistancy of the data writing to the file. All the data are interleaved as they all mixed up even all the contents are there.

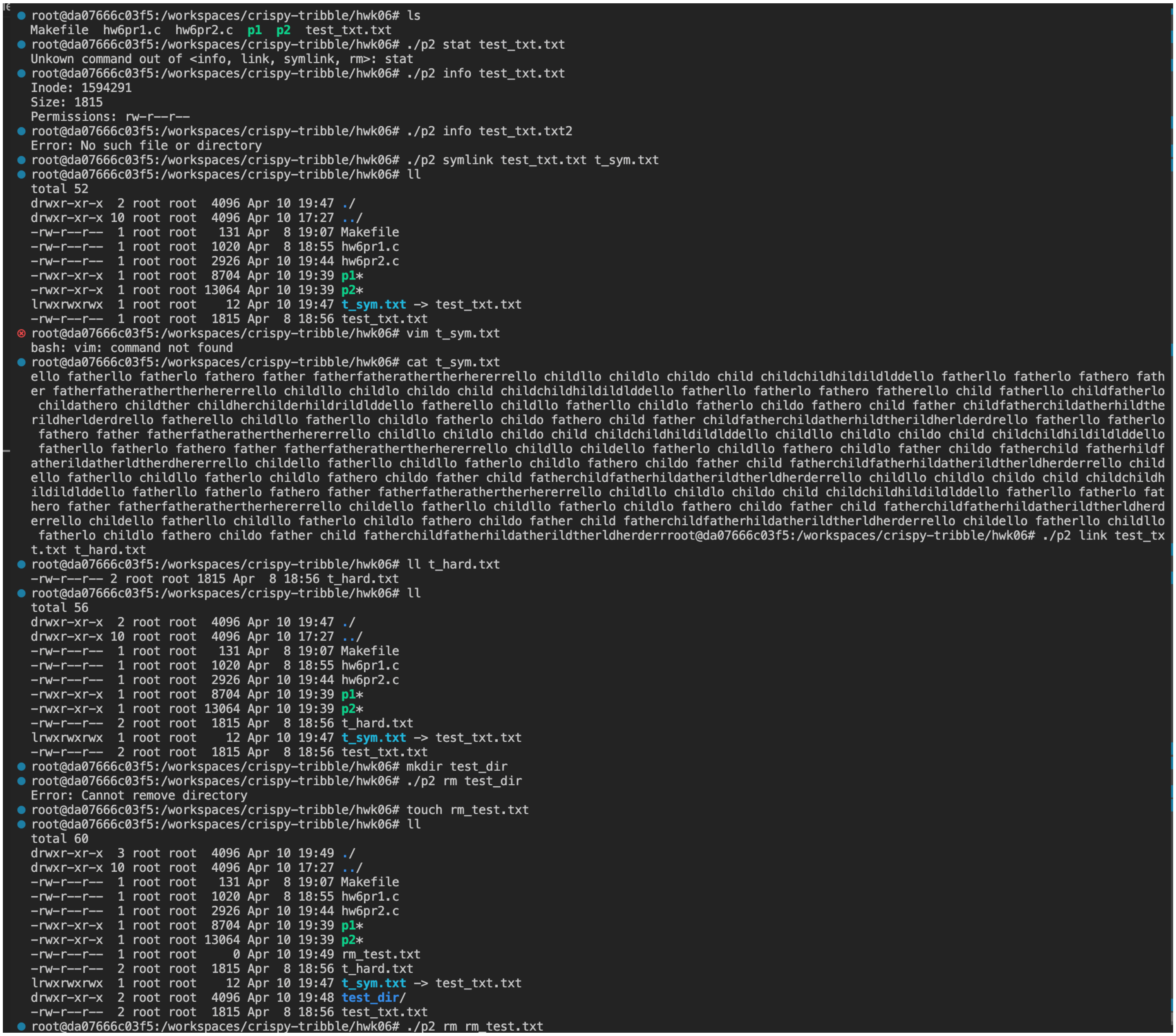
Problem 02

CMD list:

- ☑ info: `stat(file, &file_stat)` call will gather all the info(inode, permission, size) we need per file while store them in the `file_stat` struct, and we use bit operation to mask them out(for permission), and finally interpret them into **R/W/X** format;
- ☑ softlink/hardlink: use system call `symlink(src, dest)` and `link(src, dest)` to implement, set a wrapper around the user call;

- ☑ rm: you can do either `unlink()` or `remove()` to implement the remove part, also use `S_ISDIR(file_stat.st_mode)`to determine if the given file is a directroy;

Test cmd screenshot provided:



```
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ls
Makefile  hw6pr1.c  hw6pr2.c  p1  p2  test_txt.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 stat test_txt.txt
Unkown command out of <info, link, symlink, rm>: stat
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 info test_txt.txt
Inode: 1594291
Size: 1815
Permissions: rw-r--r--
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 info test_txt.txt2
Error: No such file or directory
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 symlink test_txt.txt t_sym.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ll
total 52
drwxr-xr-x  2 root root  4096 Apr 10 19:47 ./
drwxr-xr-x 10 root root  4096 Apr 10 17:27 ../
-rw-r--r--  1 root root   131 Apr  8 19:07 Makefile
-rw-r--r--  1 root root  1020 Apr  8 18:55 hw6pr1.c
-rw-r--r--  1 root root  2926 Apr 10 19:44 hw6pr2.c
-rwxr-xr-x  1 root root  8704 Apr 10 19:39 p1*
-rwxr-xr-x  1 root root 13064 Apr 10 19:39 p2*
lrwxrwxrwx  1 root root    12 Apr 10 19:47 t_sym.txt -> test_txt.txt
-rw-r--r--  1 root root  1815 Apr  8 18:56 test_txt.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# vim t_sym.txt
bash: vim: command not found
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# cat t_sym.txt
ello fatherllo fatherlo fathero father fatherfatheratherthererrello childllo childlo childo child childchildhildildlddello fatherllo fatherlo fathero fath
er fatherfatheratherthererrello childllo childlo childo child childchildhildildlddello fatherllo fatherlo fathero fatherello child fatherllo childfatherlo
 childathero childther childherchilderhildrildlddello fatherello childllo fatherllo childlo fatherlo childo fathero child father childfatherchildatherhildthe
rildherlderdrello fatherello childllo fatherllo childlo fatherlo childo fathero child father childfatherchildatherhildtherildherlderdrello fatherllo fatherlo
 fathero father fatherfatheratherthererrello childllo childlo childo child childchildhildildlddello childllo childlo childo child childchildhildildlddello
 fatherllo fatherlo fathero father fatherfatheratherthererrello childllo childello fatherlo childllo fathero childlo father childo fatherchild fatherhildf
atherildatherldtherdhererrello childello fatherllo childllo fatherlo childlo fathero childo father child fatherchildfatherhildatherildtherldherderrello child
ello fatherllo childllo fatherlo childlo fathero childo father child fatherchildfatherhildatherildtherldherderrello childllo childlo childo child childchildh
ildildlddello fatherllo fatherlo fathero father fatherfatheratherthererrello childllo childlo childo child childchildhildildlddello fatherllo fatherlo fat
hero father fatherfatheratherthererrello childello fatherllo childllo fatherlo childlo fathero childo father child fatherchildfatherhildatherildtherldherd
errello childello fatherllo childllo fatherlo childlo fathero childo father child fatherchildfatherhildatherildtherldherderrello childello fatherllo childllo
 fatherlo childlo fathero childo father child fatherchildfatherhildatherildtherldherderrroot@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 link test_tx
t.txt t_hard.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ll t_hard.txt
-rw-r--r-- 2 root root 1815 Apr  8 18:56 t_hard.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ll
total 56
drwxr-xr-x  2 root root  4096 Apr 10 19:47 ./
drwxr-xr-x 10 root root  4096 Apr 10 17:27 ../
-rw-r--r--  1 root root   131 Apr  8 19:07 Makefile
-rw-r--r--  1 root root  1020 Apr  8 18:55 hw6pr1.c
-rw-r--r--  1 root root  2926 Apr 10 19:44 hw6pr2.c
-rwxr-xr-x  1 root root  8704 Apr 10 19:39 p1*
-rwxr-xr-x  1 root root 13064 Apr 10 19:39 p2*
-rw-r--r--  2 root root  1815 Apr  8 18:56 t_hard.txt
lrwxrwxrwx  1 root root    12 Apr 10 19:47 t_sym.txt -> test_txt.txt
-rw-r--r--  2 root root  1815 Apr  8 18:56 test_txt.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# mkdir test_dir
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 rm test_dir
Error: Cannot remove directory
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# touch rm_test.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ll
total 60
drwxr-xr-x  3 root root  4096 Apr 10 19:49 ./
drwxr-xr-x 10 root root  4096 Apr 10 17:27 ../
-rw-r--r--  1 root root   131 Apr  8 19:07 Makefile
-rw-r--r--  1 root root  1020 Apr  8 18:55 hw6pr1.c
-rw-r--r--  1 root root  2926 Apr 10 19:44 hw6pr2.c
-rwxr-xr-x  1 root root  8704 Apr 10 19:39 p1*
-rwxr-xr-x  1 root root 13064 Apr 10 19:39 p2*
-rw-r--r--  1 root root     0 Apr 10 19:49 rm_test.txt
-rw-r--r--  2 root root  1815 Apr  8 18:56 t_hard.txt
lrwxrwxrwx  1 root root    12 Apr 10 19:47 t_sym.txt -> test_txt.txt
drwxr-xr-x  2 root root  4096 Apr 10 19:48 test_dir/
-rw-r--r--  2 root root  1815 Apr  8 18:56 test_txt.txt
root@da07666c03f5:/workspaces/crispy-tribble/hwk06# ./p2 rm rm_test.txt
```