

Assignment 12

- Code

```
1  #include <stdlib.h>
2  #include <stdio.h>
3  #include <string.h>
4  #include <fcntl.h>
5  #include <sys/mman.h>
6  #include <sys/stat.h>
7  #include <sys/types.h>
8  #include <unistd.h>
9
10 #define FILE_MODE 0644
11 #define COPYINCR (1024 * 1024 * 1024)
12
13 #define err_sys(fmt, ...) fprintf(stderr, (fmt "\n"), ##__VA_ARGS__)
14 #define err_quit(fmt, ...) do { err_sys(fmt, ##__VA_ARGS__); exit(1); } while(0)
15
16 int main(int argc, char *argv[])
17 {
18     int fdin, fdout;
19     void *src, *dst;
20     size_t copysz;
21     struct stat sbuf;
22     off_t fsz = 0;
23
24     if (argc != 3)
25         err_quit("usage: %s <fromfile> <tofile>", argv[0]);
26
27     if ((fdin = open(argv[1], O_RDONLY)) < 0)
28         err_sys("can't open %s for reading", argv[1]);
29
30     if ((fdout = open(argv[2], O_RDWR | O_CREAT | O_TRUNC, FILE_MODE)) < 0)
31         err_sys("can't creat %s for writing", argv[2]);
32
33     if (fstat(fdin, &sbuf) < 0) /* need size of input file */
34         err_sys("fstat error");
35
36     if (ftruncate(fdout, sbuf.st_size) < 0) /* set output file size */
37         err_sys("ftruncate error");
38
39     while (fsz < sbuf.st_size) {
40         if ((sbuf.st_size - fsz) > COPYINCR)
41             copysz = COPYINCR;
```

```

42         else
43             copysz = sbuf.st_size - fsz;
44
45             /* TODO: Copy the file using mmap here */
46             if ((src = mmap(0, copysz, PROT_READ, MAP_SHARED, fdin, f
47                 err_sys("mmap input error");
48             if ((dst = mmap(0, copysz, PROT_READ | PROT_WRITE, MAP_SH
49                 err_sys("mmap output error");
50             close(fdin);
51             memcpy(dst, src, copysz);
52             munmap(src, copysz);
53             munmap(dst, copysz);
54             fsz += copysz;
55         }
56         exit(0);
57     }
58

```

(2) Will closing the file descriptor invalidate the memory-mapped I/O?

- No. By our experiment and the reference (<https://git.kernel.org/pub/scm/docs/man-pages/man-pages.git/commit/man2/mmap.2?id=3ee0a7f07ed1bd607e598bc4477af0dea5db4084>) for the manual of mmap.



index : man-pages/man-pages.git

Linux man pages Sections 2, 3, 4, 5, and 7

[about](#) [summary](#) [refs](#) [log](#) [tree](#) **commit** [diff](#) [stats](#)

path: [root/man2/mmap.2](#)

author Elliott Hughes <enh@google.com> 2018-08-26 13:14:33 +0200
committer Michael Kerrisk <mtk.manpages@gmail.com> 2018-08-26 13:14:33 +0200
commit [3ee0a7f07ed1bd607e598bc4477af0dea5db4084](#) (patch)
tree [fff44a3b97cfd8fdeb5d7d2a3eebc9c51112c624](#) /man2/mmap.2
parent [dc9b6c92852bca1541ed13d730b8653d2b8a806d](#) (diff)
download [man-pages-3ee0a7f07ed1bd607e598bc4477af0dea5db4084.tar.gz](#)

mmap.2: Explicitly state that the fd can be closed

I regularly see excessive fd usage bugs (or even leaks) caused by people who think they need to keep the fd open as long as the mapping exists.

Signed-off-by: Michael Kerrisk <mtk.manpages@gmail.com>

Diffstat (limited to 'man2/mmap.2')

```
-rw-r--r-- man2/mmap.2 2
```

1 files changed, 2 insertions, 0 deletions

```
diff --git a/man2/mmap.2 b/man2/mmap.2
index 3d3d1882f9..7ca39ff5d6 100644
--- a/man2/mmap.2
+++ b/man2/mmap.2
@@ -87,6 +87,8 @@ in the file (or other object) referred to by the file descriptor
 .I offset
 must be a multiple of the page size as returned by
 .IR sysconf(_SC_PAGE_SIZE) .
+The file descriptor can be closed immediately without invalidating the
+mapping.
```

(3) Describe your implementation in the report.

- Line 46 to Line 49

Set our source and destination by `mmap` .

- `0` means the starting address, normally set to zero for portability.
- `copysz` represents the length of the characters we want to copy.
- We only need to open the `PROT_READ` mode for the source while both `PROT_READ` and `PROT_WRITE` for the destination.
- `fdin` and `fdout` are the file descriptors of our input file and output file.
- `fsz` is the starting offset of the file.

- Line 50

Close the input file. (The spec requires)

- Line 51

Invoke `memcpy` to copy the characters from `src` to `dst` .

- Line 52 and Line 53

Unmap the region of `src` and `dst` . Note that the region is released here, not `close(fin)` or `close(fout)` .

- Line 54

Add `copysz` to the offset.