

Exercise

Write a program that uses the POSIX file mapping services to compare two files. The program receives as arguments the names of the files to compare.

Result

```
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/mman.h>
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <string.h>

void error_and_exit(char *message) {
    perror(message);
    exit(2);
}

int main(int argc, char **argv) {
    int fd1, fd2;
    int tam;
    char *org1, *org2;
    int result=0;
    struct stat bstat;

    if (argc!=3) {
        fprintf(stderr, "Usage: %s file1 file2\n", argv[0]);
        return(2);
    }

    /* Open the files in read-only mode */
    if ((fd1=open(argv[1], O_RDONLY))<0)
        error_and_exit("Error opening the file");
    if ((fd2=open(argv[2], O_RDONLY))<0)
        error_and_exit("Error opening the file");

    /* Obtain files stats like the file size in bytes, st_size */
    if (fstat(fd1, &bstat)<0)
        error_and_exit("Error calling fstat");
    tam=bstat.st_size;
    if (fstat(fd2, &bstat)<0)
        error_and_exit("Error calling fstat");

    /* same length: could be equal */
    if (tam == bstat.st_size) {
    /* We map the files into memory */
        if ((org1=mmap(NULL, tam, PROT_READ,MAP_PRIVATE, fd1, 0)) == MAP_FAILED)
            error_and_exit("Error mapping the file");
        if ((org2=mmap(NULL, tam, PROT_READ,MAP_PRIVATE, fd2, 0)) == MAP_FAILED)
            error_and_exit("Error mapping the file ");

    /* We compare the memory map contents */
        if (memcmp(org1, org2, tam)==0)
            result=1; /* are equal */

    /* We remove the memory maps */
        munmap(org1, tam);
        munmap(org2, tam);
    }

    if (result)
        printf("Files are equal\n");
    else
```

Tema: 4

Este material se comparte con licencia Creative Commons Reconocimiento no Comercial-Compartida-Igual 3.0. España.



```
printf("Files are different\n");

/* Close all the file descriptors */
close(fd1);
close(fd2);
return(result);
}
```