# Exercise 6-5: file_write.c

Maximilian Fernaldy - C2TB1702

Note: some links and other HTML-related objects may not work in pdf form. Consider reading the webpage format of the report here.

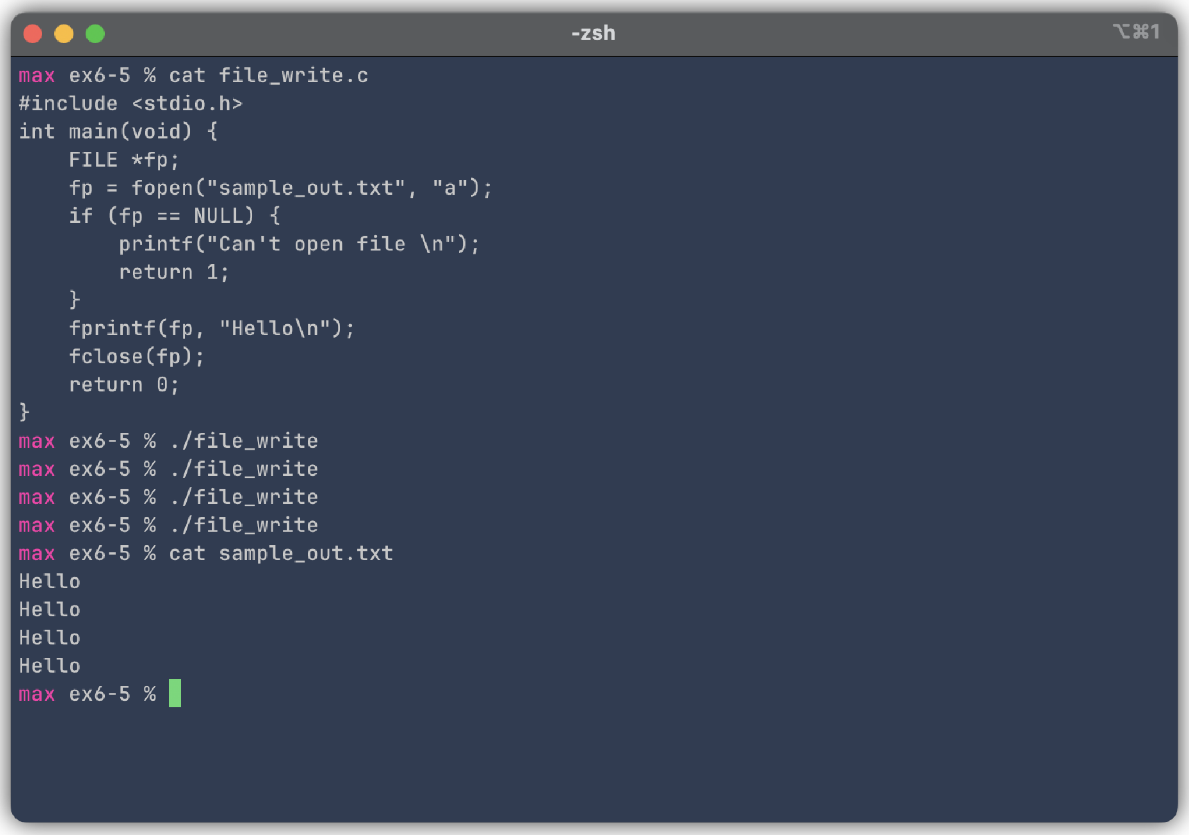
```c
#include <stdio.h>
int main(void) {
    // create file pointer
    FILE *fp;
    fp = fopen("sample_out.txt", "a");

    // handle opening error
    if (fp == NULL) {
        printf("Can't open file \n");
        return 1;
    }

    // print formatted string to file
    fprintf(fp, "Hello\n");

    fclose(fp);
    return 0;
}
```

As previously explained in report 6-4, the file access flags determine how data is written to the file. If we use `"a"` like above, new data will be written to the end of the file.

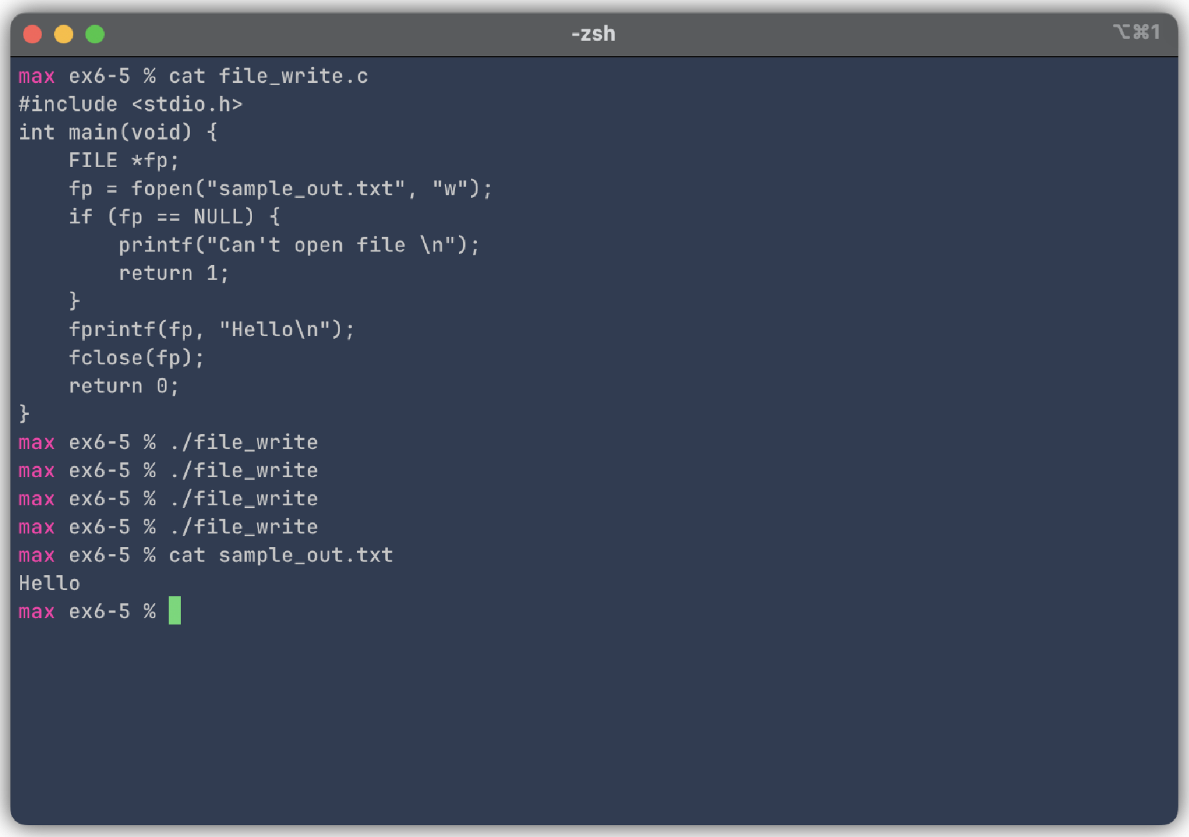


```
max ex6-5 % cat file_write.c
#include <stdio.h>
int main(void) {
    FILE *fp;
    fp = fopen("sample_out.txt", "a");
    if (fp == NULL) {
        printf("Can't open file \n");
        return 1;
    }
    fprintf(fp, "Hello\n");
    fclose(fp);
    return 0;
}
max ex6-5 % ./file_write
max ex6-5 % ./file_write
max ex6-5 % ./file_write
max ex6-5 % ./file_write
max ex6-5 % cat sample_out.txt
Hello
Hello
Hello
Hello
max ex6-5 %
```

Running the program repeatedly appends new data to the text file

It needs to be noted that `fprintf()`, just like the behavior of `printf()`, does not print new lines automatically at the end of the string (unlike `printf()` in Python, for instance). We need to append the newline character manually. This is very much intentional, and allows us to write in the same line if we'd like.

In comparison, opening the file in write mode (`"w"`) destroys previous data and rewrites the string from scratch in the file. As a result, however many times we run the program, there will only be one "Hello" followed by a blank line in the text file (the blank line is not visible in the `cat` output because it is used to carriage-return the shell prompt).



```
max ex6-5 % cat file_write.c
#include <stdio.h>
int main(void) {
    FILE *fp;
    fp = fopen("sample_out.txt", "w");
    if (fp == NULL) {
        printf("Can't open file \n");
        return 1;
    }
    fprintf(fp, "Hello\n");
    fclose(fp);
    return 0;
}
max ex6-5 % ./file_write
max ex6-5 % ./file_write
max ex6-5 % ./file_write
max ex6-5 % ./file_write
max ex6-5 % cat sample_out.txt
Hello
max ex6-5 %
```

Only one line of "Hello" is retained