

Lab Worksheet 06 - File I/O

Question 1: Code analysis

Compile and run the following program:

```
#define _XOPEN_SOURCE 700
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
#include <sys/types.h>
#include <sys/wait.h>

int main (void) {
    int fd1, fd2, fd3;

    if ((fd1 = open ("../file1", O_RDWR|O_CREAT|O_TRUNC, 0600)) == -1)
        return EXIT_FAILURE;
    if (write (fd1, "abcde", strlen ("abcde")) == -1) /* A */
        return EXIT_FAILURE;
    if (fork () == 0) {
        if ((fd2 = open ("../file1", O_RDWR)) == -1)
            return EXIT_FAILURE;
        if (write (fd1, "123", strlen ("123")) == -1) /* B */
            return EXIT_FAILURE;
        if (write (fd2, "45", strlen ("45")) == -1) /* C */
            return EXIT_FAILURE;
        close(fd2);
    } else {
        fd3 = dup(fd1);
        if (lseek (fd3, 0, SEEK_SET) == -1) /* D */
            return EXIT_FAILURE;
        if (write (fd3, "fg", strlen ("fg")) == -1) /* E */
            return EXIT_FAILURE;
        if (write (fd1, "hi", strlen ("hi")) == -1) /* F */
            return EXIT_FAILURE;
        wait (NULL);
        close (fd1);
        close(fd3);
    }
    return EXIT_SUCCESS;
}
```

Question: what are the possible contents of file file1?

Using signals, modify the program so that the file always contains exactly 8 characters.

Question 2: Custom file copy

Write a program `mycp` that takes two filenames as parameters:

```
$ ./mycp <file1> <file2>
```

This program completely copies the contents of the file `file1` into the file `file2`.

There are 2 conditions required for this program to work without returning an error:

1. the file `file1` must exist and be a regular file,
2. The file `file2` must not exist.

Question 3: Value sharing via files

Write a program where the initial parent process creates a file, then creates `N` child processes and waits for their termination. Each child generates a random value `random_val`, stores it in the file and displays it before exiting. The random value is generated thus:

```
random_val = (int) (10 * (float) rand () / RAND_MAX);
```

Once all its children have terminated, the parent process reads all the values in the file, sums them up and then displays the result, and finally deletes the file.

Question 4: Extended grep function

Write a program `extended-grep` that searches a directory for files containing a given string.

The program will be called as follows:

```
$ extended-grep <expr> <path>
```

with `expr` the string of characters sought, and `path` the directory path that contains the files to scan

`extended-grep` reads the contents of all regular files in `path`, displays the name of each file that contains the search string, or displays "search unsuccessful" if `expr` is not present in any of the regular files in the directory.

N.B: You can use the `strstr` function of the `<string.h>` library to find if a string is present in another string.