

# Grup 22: COMPROC 2

December 15th 2022

## Makefile

Create a single Makefile that allows generating all the programs at the same time and also each of them separately. Add a rule (clean) to remove all the binaries and/or object files.

## Error control and Usage

All programs must check for errors on a system call (except for write), check the number of input parameters and define the *usage()* function.

## Exercise 1

Write a program named `add_integers.c` that receives as parameter a file name. The file contains integer numbers in internal machine representation format (binary format) with no separation. The program has to read the numbers from the file and add them all up. The result of the addition has to be written to the terminal **and** to a file called `addition.txt`. If `addition.txt` does not exist, it has to be created with read and write permissions for the user, and if it already exists we want the final content of the file to be the result of the latest execution. A test input file called `numbers.txt` is provided. For example:

```
$ ./add_integers numbers.txt
55
$ cat addition.txt
55
```

## Exercise 2

Write a program named `count_lines.c`. The program has to mimic the behaviour of the following command:

```
$ cat <parameter1> | grep <parameter2> | wc -l
```

Where `<parameter1>` and `<parameter2>` are parameters of the program. The program has to create three child processes and as many unnamed pipes as necessary to communicate information across the child processes. The first child mutates to execute `cat` over the file name specified as the first parameter of the program. The second child mutates to execute `grep`, which searches on the output of `cat` the word that the program receives as second parameter. The third child mutates to execute the command `wc -l` over the output of `grep` and writes its output to the standard output. The parent process waits until all child processes finish and

then ends its execution. An execution example that counts how many lines contain the word `execlp` within the file `count_lines.c`:

```
$ ./count_lines count_lines.c execlp
3
```

## Submission

A single file `comproc2.tar.gz` that contains the Makefile and the two programs:

```
tar -czvf comproc2.tar.gz Makefile *.c
```