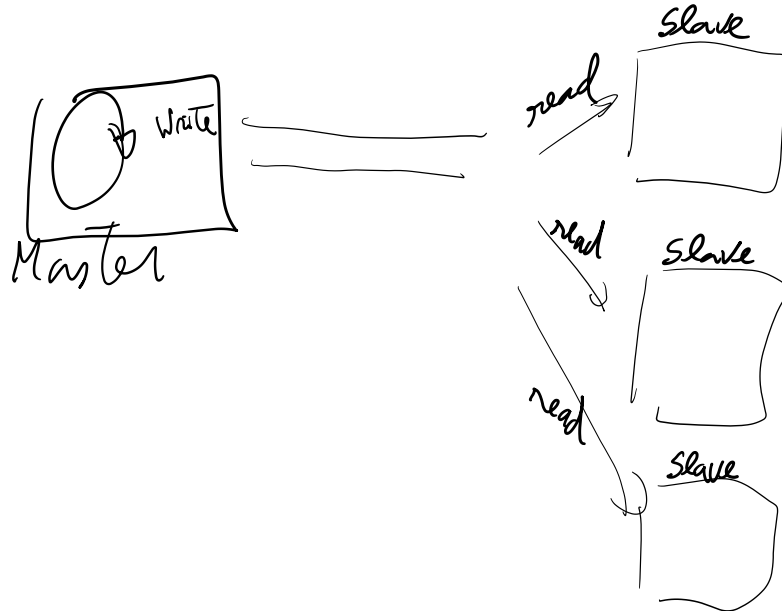


# System Programming

## 5<sup>th</sup> Laboratory (17 ... 20 March 2020)

In this laboratory students will implement a system that counts how many prime numbers there are in a



sequence of integers.

This system is composed of a master process (that generates a sequence of random numbers and writes those numbers on a pipe) and slaves processes (that continuously read integers from a pipe and verify if those numbers are prime).

The master receives as arguments (argv), the number of slaves and how many random numbers should be generated.

The master process will generate positive random numbers (lower than 99999) in a sequential manner and write them to pipe.

Each slave will read from the pipe a number and verify if it is prime.

Each slave, right before exiting should print the number of prime values read.

**The slave processes do not know how many numbers will be generated.**

### I

Implement the described system using multiple processes (created from the master).

After the master has created all the random number it will write a negative number to the pipe.

### II

Modify the previous program so that the master does not write the negative number, but closes the pipe.

**NOTE: every process reading from a pipe is notified (read returns 0) when that pipe is last closed for writing (i.e. no more processes are writing to it).**

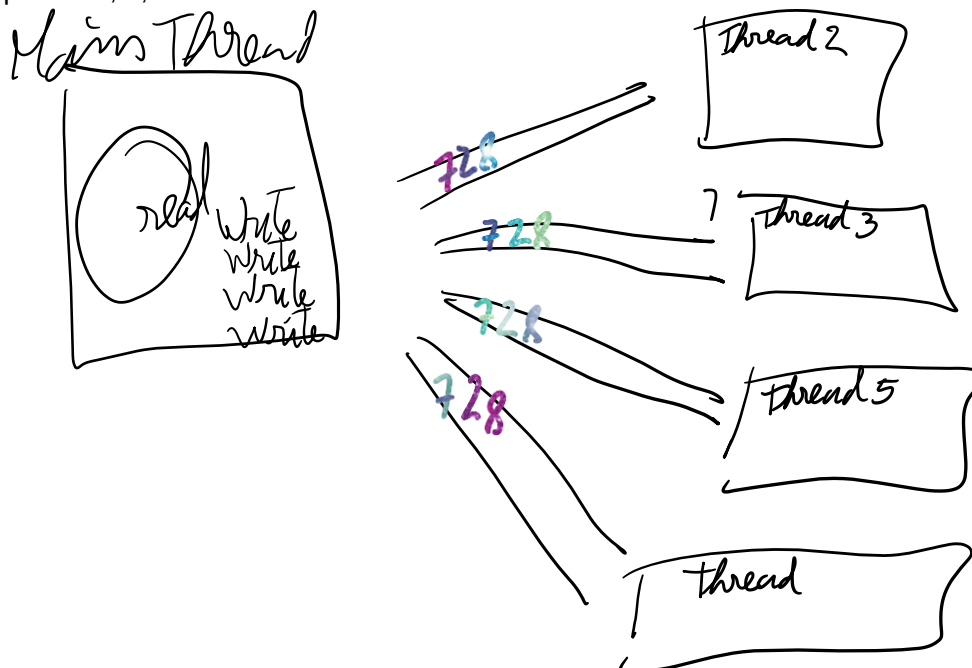
### III

Modify the previous programs so that the slaves do not print the count of prime values, but send such value to the master through another pipe.

The Master will read the results from each process, sum them and print the total count.

### III

Write a program that reads from the keyboard several integers and verifies whether these number are multiples of 2, 3, 5 or 7.



The main thread of this program reads from the keyboard integers and terminates when a negative number is read. The values read are sent into 4 different pipes.

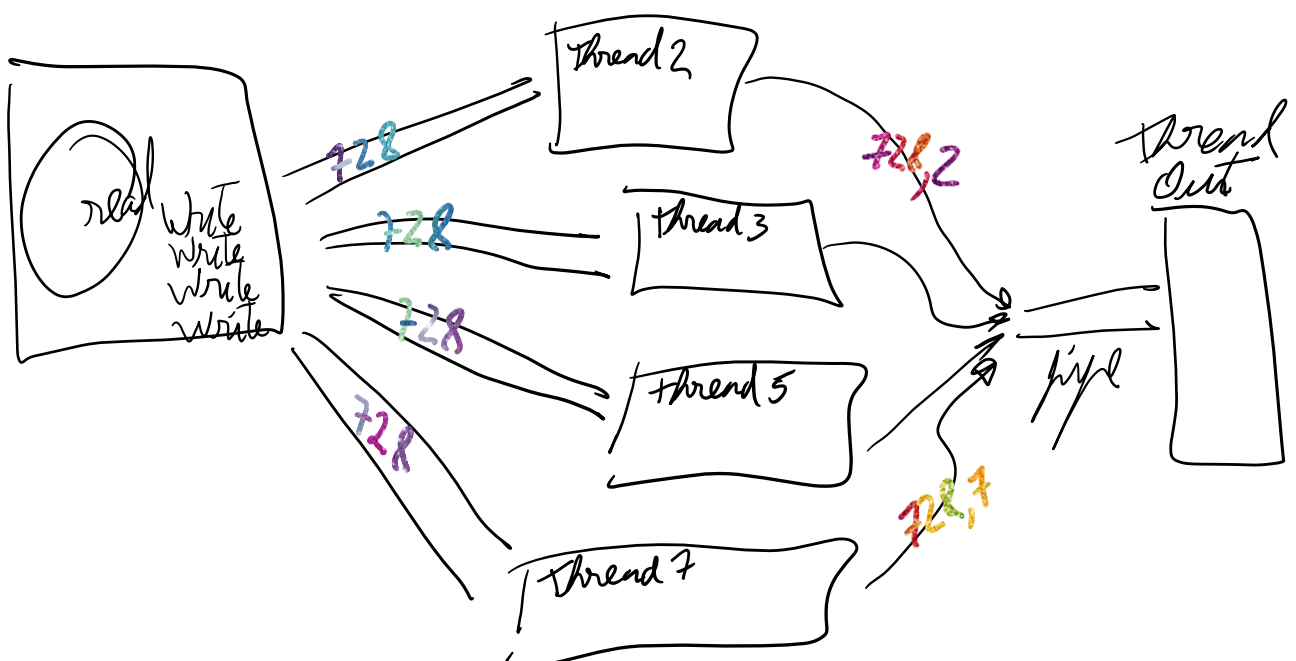
Each of the pipes is read by one thread responsible for verifying whether that number is multiple or not:

- one thread verifies if the numbers are multiples of 2
- one thread verifies if the numbers are multiples of 3
- one thread verifies if the numbers are multiples of 5
- one thread verifies if the numbers are multiples of 7

Each thread prints whether a certain number is multiple of 2,3,5 or 7.

### IV

Modify the previous program so that another thread receives information about the processed number and whether such number is multiple of 2, 3, 5 or 7.



## REFERENCES

<http://tldp.org/LDP/lpg/node7.html>

<http://beej.us/guide/bgipc/output/html/multipage/index.html>

<https://fenix.tecnico.ulisboa.pt/downloadFile/845043405486513/06-Sun-ProgGuide-IPC.pdf>