Multi-Task Programming - Assignment 2

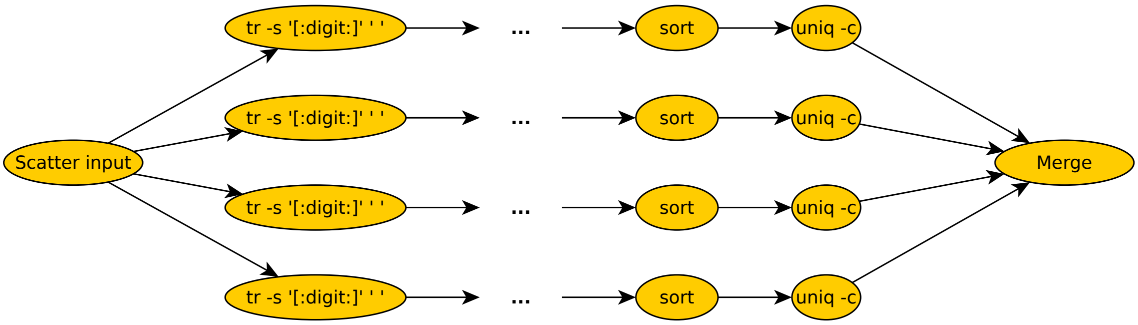
Parallel word count with pipelines

The purpose of this assignment is to implement a program in C or C++ that computes the absolute frequency of words in a text file. The following pipeline can be used for this purpose :

```
cat big.txt | tr -s '[:digit:]' ' ' | tr '[A-Z]' '[a-z]' \
    | tr -s '[:punct:]' ' ' | tr -s '\n\f\t\r ' '\n' | sort | uniq -c
```

1. (10 points) By using only system call presented in this course, write a program `pipeline.c` that executes this pipeline for a given text file. This program has to take as input a text file an write on the standard output the output of the last command.

The final program has to be designed to take advantage of all available processors to reach the best possible performance by using UNIX processes and pipes. This program has to implement the following architecture :



For this purpose, you can use the following system calls : `fork()`, `exec()`, `dup2()`, `pipe()`, `write()` and `read()`.

The next programs have to take as input a text file and write on the standard output an alphabetically ordered list of tokens along with their absolute frequency. For example:

```
$ ./freq file1
amet 9
duis 4
ipsum 5
lorem 2
paratur 2
sed 1
```

2. (6 points) With the help of `merge_sum.cpp` (that is be compiled with the command `g++ merge_sum.cpp -std=c++14`) write a program `parallel_pipeline.c` that executes exactly 2 pipelines in parallel.
3. (4 points) Write a third program `n_parallel_pipeline.c` that generalizes this previous program by executing n parallel pipelines, where the value of n has to be provided as the second argument.

The evaluation of your assignment will take into account the following points : computation time, memory usage, correctness, source code structure, and the good use of CPUs available. Your implementation will be tested on a 16 cores workstation.

Modalities

This assignment is to be made in groups of at most *two* persons.

All the source files, a `Makefile`, and a `README` (enclosed in a `.tar.gz` or `.tgz`) have to be sent by e-mail to `julien.roland@isen-lille.fr` no later than **Thursday, February 22, 2018 at 08:30**. (Please mention as subject of the email `MULTITASK PARALLEL ASSIGNMENT 2 2018`) The archive has to be named as follows: `<Lastname1Firstname1>_<Lastname2Firstname2>.tar.gz` (e.g. `ThompsonKen_RitchieDennis.tar.gz`).

The source files have to be well structured (pay attention to variable names and the use of functions), and well commented. You have to check that your program can be executed on Linux (Please mention the Linux distribution and the version of the compiler in the `README`).

The late penalty for this assignment is i points, where $i \in \mathbb{N}^*$ is the number of hours you are late.