Fundamentos de Sistemas de Operação

MIEI 2017/2018

Homework Assignment 2

Deadline and Delivery

This assignment is to be performed ***individually*** by each student – any detected frauds will cause failing the discipline. The code has to be submitted for evaluation via the Mooshak system (`http://mooshak.di.fct.unl.pt/~mooshak/`) using each student's individual account -- the deadline is ***23h59, November 3rd, 2017 (Friday)***.

Description

The goal of this assignment is to implement a *concurrent string search* in a file to report how many times a *string* appears in it. Given the string to be searched, the file's name, and the number of threads to use, the program produces the total number of occurrences of that string in the text. For instance, assuming your program is called "mygrep",

```
$ cat news.txt
Ronaldo is FIFA's best footballer of 2017. Ronaldo wins this title for the 5th time.
Ronaldo's family joined him for the celebration.
$ ./mygrep Ronaldo news.txt 2
Number of times the string "Ronaldo" occurs in news.txt: 3
./mygrep 'best footballer' news.txt 2
Number of times the string "best footballer" occurs in news.txt: 1
```

Multithreaded string count

The goal to develop a concurrent program is to identify which parts of a program benefit from dividing the work among a set of concurrent executing threads (e.g. to speed up its execution by executing in parallel) producing the same result as a sequential version of that program. In this case, your program is requested to divide the work of searching a particular string in the file's text and counting the number of times that string appears in it. To support this goal, besides producing the correct output, your program has to obey the following ***requirements***:

- The file is *totally mapped into memory*, for faster processing. You may assume that there is enough memory for mapping the entire file.
- Your program has to create as many *Posix threads* as specified as argument in the command line, and the work has to be *evenly* divided among them.
- You should *not use mutexes nor semaphores* in your code.

Additional Hints:

- One approach is to put each thread working and counting on a distinct part of the file.
- Use the following function to print the result:

```
void
print_result(const char* filename, const char* string, const unsigned int result) {
  printf("Number of times the string \"%s\" occurs in %s: %u\n",
                                                string, filename, result);
}
```

Bibliography

[1] Sections about Concurrency and Virtualization of the recommended book, "Operating Systems: Three Easy Pieces Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau"

[2] Slides of the theoretical classes and laboratory classes 4 and 6 (available from CLIP)