Text of exercise #1

Matteo Corain S256654

Laboratory #12 – System and device programming – A.Y. 2018-19

A web application orchestrates and verifies the behavior of several web servers delivering services to world-wide users. Each server is associated to one single file system directory. Each directory contains several files (one for each geographical area, e.g., Europe, USA, Japan, etc.). Each file stores accesses to that specific server coming from that specific geographic area.

The server directory list, is stored in a "server list" file, which has the following format:

```
41
c:\Windows\webServer\aragorn
c:\Windows\webServer\gandalf
c:\Windows\webServer\hobbit\frodoBeggins
c:\Windows\webServer\legolas
...
```

The first 32-bit integer specifies the number of servers (variable numberOfServers), that is, directories, manipulated by the application. Each subsequent fixed length record (spaces are eventually added when a field is shorter) specifies the name of a directory containing all word-wide accesses to the corresponding server using several "access" binary files.

Each "access" binary file has the following format:

```
178.001.192.033   goodguy    2017/01/14:13:55:36   00:01:23
034.052.001.033   badguy     2017/01/06:14:04:10   00:02:25
034.052.001.037   badguy     2017/01/14:14:04:10   00:12:38
034.052.177.048   okguy      2017/01/11:14:32:00   00:29:05
178.001.192.041   nastyguy   2017/01/13:18:29:01   00:11:52
...
```

Where each server request is represented by a record with 4 fixed-length fields (again spaces are added when a field is shorter). The first two fields indicate the IP address of the host and the name of the user from which the access to the server has been performed. The last two fields indicate the data and the time of the last access, and its time duration (hours, minutes, and seconds).

Write a Windows 32 application able to maintain up-to-date access information and to statistically evaluate this information. The application receives one file name and an integer value on the command line:

- The file stores the "server list" in binary form. Recall that each directory is associated to one server, and that all files included is such a directory store accesses to that server. Moreover, directories are flat, i.e., they do not have sub-directories.
- The integer indicates the number of threads that the Web server application has to run. Each thread has to iterate through the following operations:
  - It generates 3 random numbers, n1, n2 and n3, in the range $[0, 1.0[$.
  - It awaits (int) (n1 * 100) seconds.
  - If n2 \in [0, 0.5[the thread has to act as a "Reader". In this case, the thread reads all files stored in the directory associated to server number (int) (n3 *

`numberOfServers)`, and it computes and prints-out (on standard output) the total connection time for that server and the date and time of the last connection.

- o If `n2 \in [0.5, 1.0[` the thread has to act as a "Writer". In this case, the thread modifies all files stored in the directory associated to server number (int) (n3 * numberOfServers) by randomly modifying the access date and time and/or the access duration.

Notice that all accesses to each web server directory have to be done in mutual exclusion adopting the "First Readers and Writers" (with Readers preference) paradigm. In other words, all readers can perform their operations concurrently, but each writer must modify each single directory in mutual exclusion with all other threads possibly working on the same directory. More explicitly the "First Readers and Writers" paradigm (with Readers preference) protection has to be enforced on single (but entire) directories.