System and Device Programming

Examination Test – Programming Part
24 January 2014

Examination Time: 1h 45min. Evaluation. 18 marks.
Textbooks and/or course material allowed.
The final mark is the sum of the 1st and the 2nd parts; it cannot be refused (no retry for marks ≥ 18*).*

A "product" file contains information for each car model marketed in a country by the car industry.
For each car model the file includes a single line. Each line reports the manufacturer name, the car name, and a file name. This file stores the selling results of all seller branches for that car model. The following is a correct example a the "product" file:

```
FIAT 500 file500.txt
BMW serie1 bmwS1.txt
...
```

Which indicates that for the `FIAT 500` car the selling data are stored in `file500.txt`, etc.
Notice that the file is stored in binary form and records have a fixed-length.

For each car, a "trade" file stores, on each single line, the branch identifier, the selling date, and the selling price. The following is a correct example of the file named `file500.txt`:

```
fiatIt001 04.01.2014 15131
fiatIt006 04.01.2014 17000
dddXYZ 21.01.2014 13245
...
```

Which indicates that for the `FIAT 500`, branch `fiatIt001` sold a car the 4th of January at 15132 euros, etc. As for the "product" file, also the "trade" file is stored in binary form with records of fixed-length.

Write a C program in the MS Visual Studio environment satisfying the following specification:

- The program receives on the command line three parameters: The name of the input "product" file, an integer value `N`, and an output file name.
- The program runs `N` threads. Each thread is in charge of:
  1. Reading a single line from the "product" file, and the entire corresponding "trade" file.
  2. Evaluating some statistics on the sales figures, such as: The number of cars sold for that model, the total amount of money made for that model, the branch name which sold more models, and the date in which more models have been sold.
  3. Storing those data on the file whose name has been specified as third parameter on the command line. Each set of statistics has to be stored together with the thread identifier, the car producer and car name.
  4. Moving back to step 1, until all rows in the "product" file have been processed.
- When the entire "product" file has been processed and all `N` threads stopped running, the main thread has to collect their termination, closing the "product" file, and terminate itself.

Notice that all threads read lines from the same "product" file and write their results on the same output file. Outputs from different threads do not have to be interleaved (i.e., having one line from one thread and another from another thread is not allowed).