System and Device Programming

Examination Test – Programming Part
07 July 2017

Examination Time: 1h 45min. Evaluation. 18 marks.
Textbooks and/or course material allowed.

A polynomials of degree n of a single variable (indeterminate) v can be represented as:

$$c_0 + c_1 \cdot v^1 + c_2 \cdot v^2 + c_3 \cdot v^3 + \cdot + c_n \cdot v^n$$

A set of polynomials is stored in a binary file with the following format. Each sequence of $n + 2$ consecutive values represent a polynomial: The first real number is the value of the variable v, the next n values are coefficients (of terms of increasing power), the final number is the value of the polynomial evaluated in v and, at the beginning, is always equal to 0.0. For example, the 3 polynomials

$$
\begin{array}{rrrrrrrr}
112.23 & + & 6.34 \cdot v^1 & - & 4.45 \cdot v^2 & + & 4.56 \cdot v^3 \\
17.67 & + & 4.67 \cdot v^1 & - & 33.24 \cdot v^2 & + & 14.67 \cdot v^3 \\
123.45 & - & 89.99 \cdot v^1 & + & 34.56 \cdot v^2 & + & 5.897 \cdot v^3 \\
\end{array}
$$

are stored in a binary file equivalent to the following ASCII form:

```
3
-124.342  112.23    6.34    -4.45    4.56    0.0
  34.663   17.67    4.67   -33.24   14.67    0.0
   0.004  123.45  -89.99    34.56   5.897    0.0
```

where the first integer number indicates the degree of the polynomials and all other records represent polynomials. Notice again that for each record the first number is the value assumed by the variable v, and the last one is always zero. Moreover, all values are stored as 32-bit values.

A Windows 32 application has to evaluate all polynomials reported in the file for the value of the variable v with the following specifications:

- The program receives a file name (a string) on the command line.
- After reading the first value (the degree n) from the file, the program has to run n threads, and then await for their termination.
- Thread number i (with $i = [1, n]$) will be in charge of computing the term of degree i for all polynomials. More specifically:

  - For each record, thread i has to read the values of the variable v, and the coefficient c_i from file. Then, it must compute the value of the term $c_i \cdot v^i$. Suppose the program will run on a hardware platform where only additions and multiplications can be performed. Then, all power v^i will have to be explicitly computed by the threads as $v \cdot v \cdot v \ldots$ (i-times).
  - All threads have to proceed one record at a time, i.e., they have to **await** for each other **after** each single term computation. Then, when all terms have been computed the thread which has been **faster** to compute its term (i.e., the one that has finished **before** all others) has to compute the sum of all terms, add to this value the constant, and store the result in place of the corresponding zero value at the end of the current record. After that, all threads can move on to the next polynomial.

    Notice that at the end of the process, the file will be:
    ```
    3
    -124.342  112.23    6.34    -4.45    4.56    -8835818.819
      34.663   17.67    4.67   -33.24   14.67      571223.189
       0.004  123.45  -89.99    34.56   5.897       123.09059
    ```

Please pay particular attention to the following issues: Data structure, file manipulation, and thread synchronization.