**Homework 4**

**Premise:** This homework is optional but warmly advised to check the understanding of the student lecture-by-lecture. It should be used to improve the student preparation during the course. All the answers must be properly and clearly explained.

**Exercise 1:** This exercise refers to the example studied during the lecture of last Thursday 20/10/2016, i.e. “evaluation of a parallel program on a single-CMP architecture”. The initial computation consists in a Prod → Q → Cons pipeline where Q needs to be internally parallelized, and the system is executed on a single-CMP architecture.

- Both for the farm and the map parallelization of Q, study the under-load memory latency cost model with the original architectural specifications, i.e. the memory sub-system is composed of four mutually interleaved macro-modules directly interconnected with the four MINFs of the CMP;
- Do the same analysis at the previous point by changing the architectural specification as follows: the memory sub-system consists of four macro-modules with a sequential organization.

In the analysis of the under-load memory latency cost model, you have to determine the values of the \((R_{Q \cdot 0}, p, T_p, T_S)\) parameters and to solve the client-server system of equations according to the M/D/1 queuing model.

**Exercise 2:** Explain in which way the abstract CC architecture (with centralized GC and GSK) is implemented in a concrete directory-based architecture. Is the GSK implemented by the union of the LSKs of all the system PEs?

**Exercise 3:** Consider the following computation executed on a directory-based cache coherent multiprocessor:

shared S;

P:: { while (true) do { S = f (R, S); notify (go); < local comp. > } } 

Q:: { while (true) do { wait (go); S = g (V, S); < local comp. > } } 

Is the following semantics (represented informally by the figure below) respected owing to the cache coherence invalidation-based mechanism and the use of synchronous store instructions?

<table>
<thead>
<tr>
<th>P</th>
<th>S1 = f (R, S0)</th>
<th>...</th>
<th>S3 = f (R, S2)</th>
<th>...</th>
<th>S5 = f (R, S4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>S2 = g (V, S1)</td>
<td>...</td>
<td>S4 = g (V, S3)</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

S0, S1 and S2 denote the sequence of values assumed by the shared variable S.