<table>
<thead>
<tr>
<th>Kurskod</th>
<th>Provkod</th>
<th>Tentamensdatum</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT030A</td>
<td>T101</td>
<td>2018-10-30</td>
</tr>
</tbody>
</table>

Kursnamn  | Datateknik AV, Inbyggda datorsystem och realtidstillämpningar...

Provnamn  | Skriftlig tentamen

Ort       | Sundsvall

Termin    | H18

Ämne      | Datateknik
Written examination

DT030A, Networked embedded computer systems with real-time applications

Date: 2018-10-30

The written exam is planned to be checked and graded within 15 working days.

To achieve full score on the assignments you have to thoroughly explain every part of your solutions.

No aids allowed.

1. (3p)
   Explain the main difference between hard, soft and firm real-time systems.

2. (2+2p)
   Race conditions is something we want to avoid occurring in our solutions and especially in multi-threaded solutions.

   a) What is a race condition?
   b) Present a short program example describing this kind of condition.

3. (2p)
   Explain the difference in solutions based on “lock variables” compared to “strict alternation” as a solution for mutual exclusion problems.

4. (2+2p)
   When you want to avoid deadlock in a system the solution can be to use Banker’s algorithm developed by Dijkstra 1965.
   If you study the instance below, could we approve or do we have to disallow the questions asked.

<table>
<thead>
<tr>
<th>Has</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
</tr>
</tbody>
</table>
5. (5p)
Explain and discuss differences, similarities together with pros and cons when comparing embedded system solutions built on a SoC (=System on Chip) as the Raspberry Pi compared to a specifically dedicated system solutions like the microcontroller Arduino, ASICs (=Application Specific Integrated Circuit) or FPGA (=Field Programmable Gate Array) solutions.

6. (6p)
When we are studying some classical IPC problems we have found a problem named the *The dining philosophers* that was created by Dijkstra in 1965 and later on reformulated by Tony Hoare (developer of the algorithm *quicksort* in 1959 and inventor of the *null reference* in 1965).

Explain the basic idea of this classical IPC problem and a way to manage the mutual exclusion in such an system.

7. (8p)
Suppose that the following set of jobs is given:

<table>
<thead>
<tr>
<th>Process</th>
<th>Ci (ms), worst case execution time</th>
<th>Ti (ms), period time</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>p2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>p3</td>
<td>2</td>
<td>10</td>
</tr>
</tbody>
</table>

(a) Can we assume that the processes in the set above are schedulable with Rate Monotonic Scheduling (RMS)?

(b) What is the resulting schedule (Gantt diagram) of the RMS scheduling policy with the set above?
Networked embedded computer systems with real-time applications

\[ U = \sum_{j=1}^{n} c_j \quad \quad \quad U_{\text{max}} = n(2^l - 1) \]