IN4343 – Real Time Systems
June 25th 2015, from 14:00 to 17:00

Koen Langendoen

<table>
<thead>
<tr>
<th>Question:</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>10</td>
<td>20</td>
<td>10</td>
<td>20</td>
<td>15</td>
<td>15</td>
<td>90</td>
</tr>
<tr>
<td>Score:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- This is a closed book exam
- You may use a simple calculator only (i.e. graphical calculators are not permitted)
- Write your answers with a black or blue pen, not with a pencil
- **Always justify your answers, unless stated otherwise**

The exam covers the following material:

(a) chapters 1-6, 8-9 of the book “Hard Real-Time Computing Systems (3rd ed)” by G. Buttazzo
(b) the paper “The Worst-Case Execution-Time Problem” by Wilhelm et al. (except Section 6)
(c) the paper “Transforming Execution-Time Boundable Code into Temporally Predictable Code” by P. Puschner
(d) the paper “Best-case response times and jitter analysis of real-time tasks” by R.J. Bril, E.F.M. Steffens, and W.F.J. Verhaegh
### Hyperbolic (HB) bound
\[
\prod_{i=1}^{n} (U_i + 1) \leq 2
\]

### Response Time Analysis
- **Polling Server**
  - **schedulability**
    \[
    \forall L \in D, \quad g(0, L) \leq L
    \]
  - **response time**
    \[
    R_a = C_a + \Delta_a + F_a(T_s - C_s)
    \]
    \[
    \Delta_a = \left\lceil \frac{r_a}{T_s} \right\rceil T_s - r_a
    \]
    \[
    F_a = \left\lceil \frac{C_a - C_0}{C_s} \right\rceil - 1
    \]

- **Deferrable Server**
  - **schedulability**
    \[
    \prod_{i=1}^{n} (U_i + 1) \leq \frac{U_s + 2}{2U_s + 1}
    \]
  - **response time**
    \[
    R_a = C_a + \Delta_a - C_0 + F_a(T_s - C_s)
    \]
    \[
    C_0 = \min(C_s(r_a), \Delta_a)
    \]
    \[
    \Delta_a = \left\lceil \frac{r_a}{T_s} \right\rceil T_s - r_a
    \]
    \[
    F_a = \left\lceil \frac{C_a - C_0}{C_s} \right\rceil - 1
    \]

### Elastic Model
- **utilization**
  - **∀i**
    \[
    U_i = U_{i0} - (U_0 - U_d) \frac{E_i}{E_S}
    \]
  - **where**
    \[
    E_S = \sum_{i=1}^{n} E_i
    \]
Question 1 [10 points]

To avoid the intricacies of determining the WCET of a task, the code can be transformed into a single-path equivalent utilizing a predicated execution model [Puschner:2002]. Consider a simple microcontroller without caches, execution pipelines, branch prediction units, etc.

\begin{verbatim}
1. for( i=0; i<SIZE; i++)
2.     if (A[i] < 0)
\end{verbatim}

(a) [4 points] Explain why different execution paths could result from the above code, and how many in case the SIZE of the array is set to 50.

(b) [3 points] Provide single-path code for the transformed loop body (lines 2+3) for a processor with a fully predicated instruction set.

(c) [3 points] Discuss the complication of targeting a microcontroller with a data cache, and provide predicated code that guarantees a deterministic execution trace.

Question 2 [20 points]

Given the following set of aperiodic, preemptable tasks simultaneously released at t=0

\begin{tabular}{|c|c|c|c|c|c|c|}
\hline
 & A & B & C & D & E & F & G \\
\hline
\hline
$C_i$ & 2 & 3 & 3 & 5 & 1 & 2 & 5 \\
\hline
$d_i$ & 17 & 15 & 20 & 13 & 25 & 22 & 19 \\
\hline
\end{tabular}

with precedent constraints:

A→C, B→C, B→D, C→E, C→F, D→F, D→G.

(a) [5 points] Derive the schedule produced by the Latest Deadline First (LDF) policy. Report the response times and lateness of each task.

(b) [10 points] Proof that LDF is optimal (i.e. minimizes the maximum lateness) when tasks with precedence constraints are activated synchronously (i.e. all at once). Hint: start from a non-LDF schedule and use an interchange argument to show that LDF yields a schedule with equal or better maximum lateness.

(c) [5 points] Provide a task set in which LDF outperforms (plain) EDF.

Question 3 [10 points]

\begin{tabular}{|c|c|c|}
\hline
$C_i$ & $T_i$ & $	au_1$ 3 5 \\
\hline
$\tau_2$ & 2 6 \\
$\tau_3$ & 1 7 \\
\hline
\end{tabular}

\begin{tabular}{|c|c|c|}
\hline
$C_i$ & $T_i$ & $	au_1$ 3 6 \\
\hline
$\tau_2$ & 2 7 \\
$\tau_3$ & 1 8 \\
\hline
\end{tabular}

\begin{tabular}{|c|c|c|}
\hline
$C_i$ & $T_i$ & $	au_1$ 3 7 \\
\hline
$\tau_2$ & 2 8 \\
$\tau_3$ & 1 9 \\
\hline
\end{tabular}

(i) (ii) (iii)

(a) [7 points] Which task sets are feasible under Rate Monotonic scheduling?

(b) [3 points] Which task sets are feasible under Earliest Deadline First scheduling?
Question 4  
[20 points]

<table>
<thead>
<tr>
<th></th>
<th>$C_i$</th>
<th>$D_i$</th>
<th>$T_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>3</td>
<td>5</td>
<td>8</td>
</tr>
</tbody>
</table>

(a) 2 points  State the difference between Rate Monotonic and Deadline Monotonic scheduling.

(b) 5 points  Determine if the above task set is feasible under DM scheduling.

(c) 13 points  Determine if the task set is feasible under EDF scheduling.

Question 5  
[15 points]

Consider the following periodic tasks under RM scheduling:

<table>
<thead>
<tr>
<th>$\tau_i$</th>
<th>$C_i$</th>
<th>$T_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>3</td>
<td>12</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>4</td>
<td>24</td>
</tr>
</tbody>
</table>

(a) 5 points  **Compute** the best case response times for the above task set.

(b) 10 points  Consider that task $\tau_1$ incurs an activation jitter of 1. **Draw** the optimal instants for tasks $\tau_2$ and $\tau_3$, and report the observed best case response times.

Question 6  
[15 points]

When mixing periodic and aperiodic tasks one can make use of a priority server to schedule the aperiodic tasks. Consider the following periodic tasks and aperiodic jobs (under EDF):

<table>
<thead>
<tr>
<th>$\tau_i$</th>
<th>$C_i$</th>
<th>$T_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>4</td>
<td>9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$J_i$</th>
<th>$a_i$</th>
<th>$C_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_1$</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$J_2$</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

(a) 5 points  Compute the response times for the two jobs when being served by (plain) TBS.

(b) 5 points  Compute the response times for the two jobs when being served by **optimized** TBS.

(c) 5 points  Compute the response times for the jobs when being served by CBS.