IN4343 – Real Time Systems
April 11th 2019, from 09:00 to 12:00

Mitra Nasri

<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>7</th>
<th>8</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points:</td>
<td>10</td>
<td>15</td>
<td>20</td>
<td>10</td>
<td>5</td>
<td>15</td>
<td>5</td>
<td>5</td>
<td>85</td>
</tr>
<tr>
<td>Score:</td>
<td></td>
<td></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
- Question 8 is an optional question (bonus).

- The exam covers the following material:
  
  (b) the paper “The Worst-Case Execution-Time Problem” by Wilhelm et al. (except Section 6)
  (c) the paper “Non-Work-Conserving Non-preemptive Scheduling: Motivations, Challenges, and Potential Solutions” by Mitra Nasri and Gerhard Fohler
  (d) the paper “An Exact and Sustainable Analysis of Non-Preemptive Scheduling” by Mitra Nasri and Björn B. Brandenburg
  (e) 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
| Liu and Layland (LL) bound | $U_{RM}^n = n(2^{1/n} - 1)$ |
| Hyperbolic (HB) bound | $\prod_{i=1}^n (U_i + 1) \leq 2$ |
| Response Time Analysis | $\begin{align*}
WR_i &= C_i + \sum_{k=1}^{i-1} \left\lceil \frac{WR_i + AJ_k}{T_k} \right\rceil C_k \\
BR_i &= C_i + \sum_{k=1}^{i-1} \left( \left\lceil \frac{BR_i - AJ_k}{T_k} \right\rceil - 1 \right)^+ C_k \\
w^+ &= \max(w, 0)
\end{align*}$ |
| Processor Demand | $\begin{align*}
g(t_1, t_2) &= \sum_{r_i \geq t_1} C_i \\
g(0, L) &= \sum_{i=1}^n \left\lceil \frac{L + T_i - D_i}{T_i} \right\rceil C_i
\end{align*}$ |
| CBS Server | if (exists a pending aperiodic job) then enqueue $J_k$; |
| | else if ($q_s \geq (d_s - a_k) \cdot U_s$) then |
| | \{ $q_s \leftarrow Q_s$; $d_s \leftarrow d_s + T_s$; \} |
| | $q_s \leftarrow Q_s$ |
| | $d_s \leftarrow d_s + T_s$ |
| NP scheduling | response time NP-FP |
| | pre-requisites: $D \leq T$ and preemptive-schedulable |
| | $WO_i^{(n)} = B_i + \sum_{k=1}^{i-1} \left( \left\lceil \frac{WO_i^{(n-1)}}{T_k} \right\rceil + 1 \right) \cdot C_k$ |
| | $B_i = \max\{C_j \mid \forall \tau_j, P_i < P_j\}$ |
| | $R_i = C_i + WO_i$ |
| | $WO_i^{(0)} = B_i + \sum_{k=1}^{i-1} C_k$ |
| | $\forall \tau_i, 1 < i \leq n, \forall L, T_1 < L < T_i$: |
| | $L \geq C_i + \sum_{k=1}^{i-1} \left\lceil \frac{L - 1}{T_k} \right\rceil \cdot C_k$ |
Question 1

Which task sets are feasible under Rate Monotonic scheduling?

Which task sets are feasible under Earliest Deadline First scheduling?

Question 2

Consider the following task set:

Use the processor demand criterion to demonstrate that the above task set is unfeasible under EDF scheduling.

Determine the worst-case response times of the three tasks. *Hint: draw a time line.*

Question 3

Consider the following periodic tasks and aperiodic jobs:

The system designer wants to use fixed-priority servers to schedule the aperiodic jobs. Assume the server is configured as \((C_s = 2, T_s = 5)\) and the rate monotonic scheduling is used to schedule the tasks and the server.

(a) Can a polling server be used without compromising the feasibility of the periodic tasks? Determine the worst-case response times of the two jobs.

(b) Can a deferable server be used without compromising the feasibility of the periodic tasks? Determine the worst-case response times of the two jobs.

(c) Now consider a fixed-priority scheduling. What is the highest possible priority that one can assign to this deferable server such that none of the periodic tasks miss their deadline in the example above? (your answer must include the priority of each of the tasks as well as the server.)

(d) What is a disadvantage of a polling server and how a deferable server can improve it?
Question 4 [10 points]

To schedule the following aperiodic jobs ($J_1$ and $J_2$), a constant bandwidth server with the period $T_s = 6$ and budget $Q_s = 2$ is used.

<table>
<thead>
<tr>
<th>Job</th>
<th>$a_i$</th>
<th>$WCET_i$</th>
<th>$T_i$</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>0</td>
<td>2</td>
<td>6</td>
<td>periodic</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>0</td>
<td>3</td>
<td>9</td>
<td>periodic</td>
</tr>
<tr>
<td>$J_1$</td>
<td>2</td>
<td>3</td>
<td>9</td>
<td>aperiodic</td>
</tr>
<tr>
<td>$J_2$</td>
<td>12</td>
<td>2</td>
<td>6</td>
<td>aperiodic</td>
</tr>
</tbody>
</table>

where $a_i$ denotes the arrival time of the job (or the phase of the periodic task). Note that for periodic tasks, the deadline is equal to the period.

(a) Draw the schedule from time 0 to time 18.

Question 5 [5 points]

Consider a system scheduled by a fixed-priority scheduling in which there is a polling server that has the highest priority among all periodic tasks.

(a) Generate an example of a periodic task set and an aperiodic job for which the response time of the aperiodic job is larger when it is scheduled within the polling server than when it is scheduled as a background service. Draw the schedule of your task set together with the job to justify your answer.

Question 6 [15 points]

(a) What is the advantage of fixed-priority scheduling over EDF when there might be overloads in the system?

(b) Briefly explain what is the cache-related preemption delay and how does it impact the schedulability.

(c) What is the length of critical path in the following directed-acyclic graph (DAG)? Assume that the values written on the vertices are the WCET of the code segment and assume there is an infinite number of processors.
Consider the highest-locker priority (HLP) protocol for real-time tasks that use
shared resources protected by semaphores. List one advantage of HLP over the Non-preemptive
Resource Access Protocol (NPP) and one disadvantage of HLP (in general).

List two advantage of partitioned multiprocessor scheduling over global multipro-
cessor scheduling.

Question 7 [5 points]

Answer the following multiple-choice questions. Note that some questions have fewer than 4 choices.

(a) Claim: "A deferable server may impact the schedule of the higher-priority peri-
odic tasks"
   Cause: "because it keeps its budget even if there is no aperiodic jobs and hence it may
   schedule two aperiodic jobs consecutively (one at the end of the server’s period and one at
   the beginning of the next activation of the server). It results in a long delay for the periodic
   tasks."
   (a) claim is true, cause is true  (b) claim is true, cause is false  (c) claim is false

(b) Claim: A periodic task set with $U = 1$ is never schedulable by the preemptive
rate-monotonic scheduling algorithm.
   Cause: because the Liu and Layland test will certainly reject that task set.
   (a) claim is true, cause is true  (b) claim is true, cause is false  (c) claim is false

(c) In the following demand-bound function (DBF) that has been drawn for a peri-
odic task set with two tasks, what are the WCET and deadline of the task whose period is 4?
   (hint: the other task has a period larger than 4).
   (a) $C_1 = 1, D_1 = 3$  (b) $C_1 = 2, D_1 = 3$  (c) $C_1 = 2, D_1 = 4$  (d) $C_1 = 1, D_1 = 4$

(d) Claim 1: If a task set does not pass the hyperbolic-bound test, then it will also not
pass the Liu and Layland test.
   Claim 2: If a task set is harmonic and feasible (i.e., $U < 1$), then the hyperbolic-bound
test will certainly accept it.
(a) claim 1 is true, claim 2 is true    (b) claim 1 is true, claim 2 is false
(c) claim 1 is false, claim 2 is true    (d) claim 1 is false, claim 2 is false

(e) 1 point  Assume that the following task set is scheduled using **global non-preemptive fixed-priority** algorithm with priorities $P_2 < P_1 < P_3 < P_4$ on a multicore platform with 3 cores. What is the **worst-case response time** of $\tau_1$?

(a) 1    (b) 2    (c) 3    (d) 4

**Question 8**

The following question is optional and acts as a bonus question!

(a) **5 points** Briefly explain how online preemptive fixed-priority scheduling can be implemented in a constant runtime regardless of the number of tasks and priorities in the system.