EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Mathematics and Computer Science

Examination of Real-time Systems (2IMN20)
on Wednesday, June 29^{th} 2016, 18.00h-21.00h.

First read the entire examination. There are 6 exercises in total. Grades are included between parentheses at all parts and sum up to 10 points. *Motivate all your answers.* Good luck!

1. *Multi-rate cyclic executives*

   (a) (0.5) Explain the difference between a *multi-rate time-driven AFAP* and a *multi-rate periodic* cyclic executive. *Hint*: use pictures.
   Answer: See slides `RTS.B3-Cyclic-Executive`.

   (b) (0.5) What is the advantage of a *multi-rate periodic* compared to a *multi-rate time-driven AFAP* cyclic executive.
   Answer: See slides `RTS.B3-Cyclic-Executive`.

   (c) (0.5) Consider a real-time application consisting of three non-preemptive periodic tasks; see the following table providing their characteristics. Describe a problem that may arise when using the *multi-rate time-driven AFAP* cyclic executive.

	T	D	WC
τ_1	10	3	2
τ_2	10	5	3
τ_3	20	15	5

   Answer: Multi-rate time-driven AFAP will not work, because the tasks are not activated strictly periodic. As an example, task τ_1 may be activated with an inter-arrival time that is both *shorter* and *longer* than 10, i.e. when $C_1 + C_2 + C_3 < 10$. Due to the *activation jitter* and the increased inter-arrival time, it may detect an event too late, and therefore not be able to meet the *system* deadline.

   This becomes clear when considering `RTS.D0-Water-Vessel` (slide 7). When the inter-arrival of τ becomes *longer* and τ is activated just before the critical condition becomes true, the next activation of τ may be too late to allow the system to meet its deadline D^{System}. More specific, with activation jitter AJ^{τ}, the schedulability condition becomes $T^{\tau} + AJ^{\tau} + D^{\tau} \leq D^{\text{System}}$. Compared to a strictly periodic activation, the relative deadline D^{τ} shall therefore be reduced with AJ^{τ} to maintain schedulability.

2. *Utilization bounds*
   The LL-bound ($LL(n) = n(2^{1/n} - 1)$) and HB-bound ($HB(n) : \prod_{i=1}^{n}(U_i + 1) \leq 2$) are examples of sufficient conditions for fixed-priority preemptive scheduling (FPPS).

   (a) (0.5) Give at least three assumptions on which these bounds are based.
   Answers: It concerns the basic assumptions for the analysis, e.g. deadlines equal to periods, rate-monotonic scheduling, tasks do not suspend themselves, the overhead of context switching and scheduling is ignored,and tasks are independent.

   (b) (1.0) Construct an example of three tasks using a timeline for which $HB(3) = 2$, i.e. an increase of the computation time of any task will make the task set unschedulable.

Explain whether or not the LL-bound holds for this taskset? *Remark*: A calculation does not yield points.

Answer: For an example, see exercise "HB-bound" in `RTS.B5-Analysis-1-Basics`. Unless $\forall_{1 \leq i \leq 3} U_i = 2^{1/n} - 1$, with $n = 3$, the LL-bound will not hold; see book p. 99 and exercises in `RTS.B5-Analysis-1-Basics`.

3. *Resource Access Protocols*

   (a) (1.0) Draw a timeline illustrating a transitive priority adjustment. Make sure you clearly indicate the tasks, resources, and the (un-) locking operations.
   Answer: See slides `RTS.B4-Policies-3-RAP` or Figure 7.10 in the book on p. 218.

   (b) (0.5) Can a transitive priority adjustment occur for PCP?
   Answer: The keyword is "*system ceiling*" and the answer is no. Referring to the figure in the book, PCP doesn't allow task τ_2 (with the middle priority) to access resource "*a*", because the access of task τ_3 to resource "*b*" caused the system ceiling to be set to the priority of task τ_2, which is higher than or equal to the priority of task τ_2. See also Lemma 7.7 in the book.

4. *Response-time analysis*
   Consider four tasks that are scheduled by means of FPPS, where τ_1 has highest and τ_4 has lowest priority, with arbitrary phasing and characteristics as given below.

	$WT = BT$	WD	BD	AJ	$WC = BC$
τ_1	5	3	1	1	2
τ_2	10	8	3	2	3
τ_3	18	15	1	0	1
τ_4	35	34	15	1	6

   (a) (0.5) Determine the worst-case response time of τ_4 by means of the following recursive equation.

$$x = B_i + WC_i + \sum_{1 \leq j < i} \left\lceil \frac{x + AJ_j}{WT_j} \right\rceil WC_j. \quad (1)$$

   Answer: Note that the *lowest* priority task is *never* blocked. Hence, $B_4 = 0$ irrespective of potential resource sharing. Using an iterative procedure, we find $WR_4 = 34$.

   (b) Determine the best-case response time of task τ_4

      i. (1.0) by drawing a time line with an optimal instant for τ_4.
      Answer: See Figure 1. Note that $BR_4 = 13$.

      ii. (0.5) by means of the following recursive equation.

$$x = BC_i + \sum_{1 \leq j < i} \left(\left\lceil \frac{x - AJ_j}{BT_j} \right\rceil - 1 \right)^+ BC_j \quad (2)$$

      Answer: Using $WR_4 = 34$ as initial value for the iterative procedure to determine the best-case response time BR_4, we find $BR_4 = 13$.

5. *Fixed-priority servers*
   Consider a deferrable server DS, two tasks τ_1 and τ_2, and three aperiodic requests a_1, a_2, and a_3 with characteristics as given below. Assume a rate-monotonic priority assignment of the server and the tasks and FPPS.

	$T = D$	C
S	6	2
τ_1	8	2
τ_2	10	3

	t	C
a_1	5	2
a_2	9	1
a_3	12	2

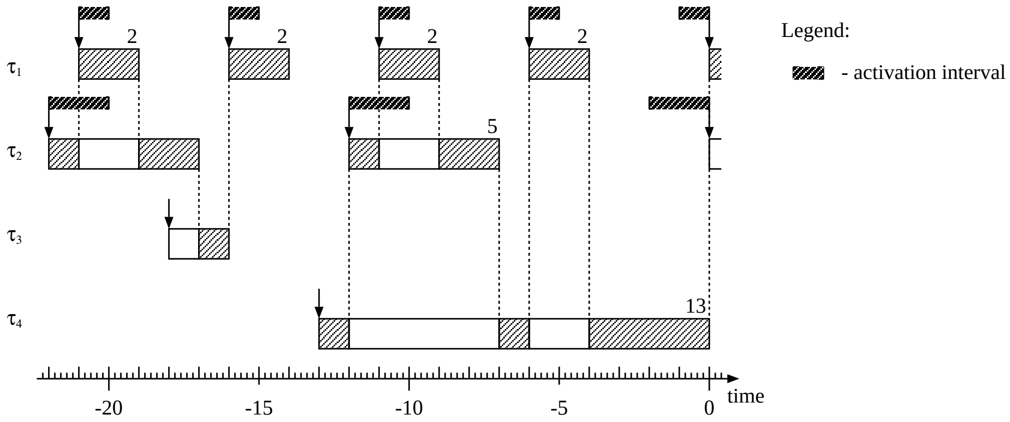

Figure 1: Timeline with an optimal instant for task τ_4.

(a) (1.0) Assume a simultaneous release of DS, τ_1, and τ_2 at time $t = 0$. Draw a timeline for $t \in [0, 18)$ for the server, the tasks, and the aperiodic requests.
Answer: See slide in `RTS.B4-Policies-2-FP-servers` or Figure 5.8 in the book.

(b) (0.5) Determine whether or not task τ_2 is schedulable under arbitrary phasing and a worst-case arrival of aperiodic requests. *Hint*: use (1) or draw a timeline.
Answer: The activation jitter AJ_S of the deferrable server S is $AJ_S = T_S - C_S = 4$. Using (1) we find that $WR_3 > D_3$.

6. *Fixed-priority scheduling with deferred preemptions (FPDS)*

   (a) The analysis for FPDS is based on a *continuous* rather than a *discrete* model.

      i. (0.5) Explain the differences between both models.
      Answer: For a continuous model, all task characteristics and task activations are taken from a dense domain, e.g. the real or rational numbers. As a consequence, between any two moments in time, there exists another moment in time. For a discrete model, the values are taken from, for example, the natural numbers or the integers.
      This question was related to exercise 1 in `RTS.B3-Reference-Model`.

      ii. (0.5) Explain the consequences for the analysis for FPDS.
      Answer: See `Concluding remarks` slide of `RTS.B5-Analysis-7-FPDS`.

   (b) (0.5) Describe the notion of a level-i active period in your own words.
   Answer: An interval $[t_s, t_e)$ where the cumulative pending load of tasks $\tau_1, \ldots, \tau_i$ is zero at time t_s and time t_e, and larger than zero for all $t \in (t_s, t_e)$; see for example slides of `RTS.B5-Analysis-FPPS-arbitrary deadlines`.

   (c) (0.5) The response time of all jobs in a level-i active period have to be considered to determine the worst-case response time of a task τ_i under FPDS and *deadlines at most equal to periods*. Explain in your own words why looking at the first job alone is not sufficient.
   Answer: The last sub job of a job may defer the execution of higher priority tasks, which may cause a longer response time of the next job.