technische universität dortmund
fakultät für informatik
computer science 12

Prof. Jian-Jia Chen

Real-Time Systems (SS 2014)

Exercise 1: Concept of Real-Time Systems and Static-Priority Scheduling

Discussion Date: 30, April 2014

Exercise 1.1

What are the main differences between general purpose computing and real-time computing? List some applications for different levels of supports of real-time systems.

Exercise 1.2

For real-time systems, it is important to know the maximum (worst-case) execution time of each task a priori. What are the definition and difference between the worst-case execution time and the worst-case response time? Even if the worst-case execution time of a task is given, there are several other problems that may be encountered during the design of a scheduling algorithm for a real-time system. Can you think of some difficulties? What are possible solutions?

Exercise 1.3

Suppose that the following set of jobs is given:

	J_1	J_2	J_3	J_4	J_5
a_j	0	2	8	10	15
C_j	4	3	6	3	4
d_j	6	8	20	14	22

(a) What is the resulting schedule of the shortest-job-first (SJF) scheduling policy?

(b) What is the resulting schedule of the earliest-deadline-first (EDF) scheduling policy?

(c) What is the average response time of SJF and EDF, respectively?

(d) Mr. S claims that SJF is optimal for his system, and Miss E claims that EDF is optimal for her system. Is it possible that both of them are correct? Please make their descriptions more clear.

Exercise 1.4
Suppose that we are given the following 3 sporadic real-time tasks with implicit deadlines.

	τ_1	τ_2	τ_3
C_i	1	2	3
T_i	4	6	10

(a) What are their priority levels? Is the rate-monotonic (RM) schedule feasible?

(b) What happens if we change the minimum inter-arrival time of task τ_3 from 10 to 8.

Exercise 1.5
Explain how to use the time-demand schedulability test to prove that rate-monotonic scheduling is an optimal static-priority scheduling policy (with respect to schedulability).

Challenge 1.6
Mr. Smart suggests the following schedulability test of static-priority scheduling for sporadic real-time tasks, as defined in the course. He claims that task τ_i can meet its its relative deadline under the static-priority scheduling if and only if the following mixed-integer linear programming has a solution.

$$C_i + \sum_{j=1}^{i-1} n_j \cdot C_j \leq t \tag{1}$$

$$n_j \cdot T_j \geq t \qquad \forall j = 1, 2, \ldots, i-1 \tag{2}$$

$$n_j \in \mathbf{N} \qquad \forall j = 1, 2, \ldots, i-1 \tag{3}$$

$$0 < t \leq D_i, \tag{4}$$

where t is a positive variable, described in (4), and n_j is a positive integer number, described in (3). Please either explain/prove or disprove his argument.