Parallel and Distributed Real-Time Systems
EDA421/DIT171

Final exam, December 16, 2009 at 14:00 – 18:00 in the V building

---

Examiner:
Docent Jan Jonsson
Phone: 031–772 5220

Content:
The written exam consists of 4 pages (including cover), containing 7 problems worth a total of 60 points.

Grading policy:
24–35 ⇒ grade 3
36–47 ⇒ grade 4
48–60 ⇒ grade 5

Restrictions:
Books, notes or calculators are NOT allowed (only writing material and dictionaries)

Solution:
No solution provided.

Results:
Posted on the course home page on Monday, January 11, 2010 at 09:00.

Inspection:
Room 4128, Rännvägen 6 B, on Thursday, January 14, 2010 at 13:00–14:00. Inspection at another occasion could be arranged by contacting the course examiner.

Language:
Your solutions should be written in English.

IMPORTANT ISSUES

1. Use separate sheets for each answered problem, and mark each sheet with the problem number.
2. Justify all answers. Lack of justification can lead to loss of credit even if the answer might be correct.
3. Explain all calculations thoroughly. If justification and method is correct then simple calculation mistakes do not necessarily lead to loss of credit.
4. If some assumptions in a problem are missing or you consider that the made assumptions are unclear, then please state explicitly which assumptions you make in order to find a solution.
5. Write clearly! If I cannot read your solution, it is wrong.

Good Luck!

PROBLEM 1

State whether the following propositions are True or False. For each correct statement, you will be given 1 point; for each erroneous statement you will be given -1 point; if you make no statement at all, you will be given 0 points. **Quality guarantee**: The total result for this problem cannot be less than 0 points. (6 points)

a) The *jitter* of a periodic task in a schedule reflects the amount of deviation from the expected periodicity of the task's completions.

b) The Token Ring standard (IEEE 802.5) allows for 16 different message priority levels.

c) One of the so-called *Richard's anomalies* states that task completion times may increase as a result of increasing the number of processors.

d) By the term *critical instant* in schedulability analysis we mean a point in time where a task's response-time is maximized.

e) When there are mutual exclusion constraints in a system, it is impossible to find an optimal on-line scheduling algorithm (unless it is clairvoyant).

f) In the Fault-Tolerant Average algorithm for clock synchronization, the new clock value is the mean of all collected clock readings excluding the t fastest and t slowest clocks.

PROBLEM 2

The following questions are related to estimation of worst-case execution times (WCET) of tasks.

a) Explain what the term *infeasible path* refers to in the context of WCET analysis. Also explain why it is useful to identify infeasible paths. (2 points)

b) Give two examples of techniques that are used for taking real measures (not analytical estimations) of the execution times in a computer system. (2 points)

c) Explain how *low-level analysis* works in the context of WCET analysis. (4 points)

PROBLEM 3

When a communication network is used in a real-time system it is important to know the worst-case *message delay* between the sender and receiver of an interprocessor data exchange. Two important components of the message delay are the *queuing delay* and the *transmission delay*.

a) Describe briefly the underlying causes of the queuing delay in the following networks: TDMA, Ethernet and Token Ring. (3 points)

b) Describe the mechanisms that give rise to the queuing delays on a Controller Area Network (CAN) bus system. (3 points)

c) Describe how the transmission delay is calculated for a communication link. (2 points)

d) Describe two techniques for assigning deadlines to interprocess communication. (2 points)

PROBLEM 4

For static (offline) multiprocessor scheduling, one useful algorithm is Simulated Annealing (SA).

a) Describe the major principles behind the SA algorithm when used in the general sense (not necessarily for real-time scheduling). (4 points)

b) Describe how the SA algorithm can be adapted to solve the real-time multiprocessor scheduling problem (according to Tindell, Burns and Wellings). (2 points)

c) When evaluating the scheduling performance of the SA algorithm, what evaluation methodology can be used: theoretical analysis or experimental analysis? Motivate your answer! (2 points)

PROBLEM 5

Let a *DPPE task set* be a set of tasks with deadline, precedence, preemption and exclusion constraints. The DPPE schedulability problem — that is, the problem of determining (hard real-time) schedulability on a uni-processor for a DPPE task set — is NP-complete.

a) Describe the main differences between the problem classes P and NP. (2 points)

b) Describe the four-step procedure used for proving that a decision problem (such as the DPPE schedulability problem) is NP-complete. (4 points)

c) The B&B algorithm by Xu and Parnas (as studied in homework assignment #1) solves the optimization problem of finding the schedule (if one exists) with the minimum possible task lateness for a DPPE task set on a uni-processor. Given the NP-complete DPPE schedulability problem, show that the corresponding DPPE *optimization* problem (as solved by Xu and Parnas' algorithm) is NP-hard. (4 points)

PROBLEM 6

One method for analyzing the performance of a system that experiences overload is to use the concept of *competitive factor*.

a) Give the formal definition of competitive factor. (4 points)

b) What is the best achievable competitive factor for an on-line scheduler according to Baruah *et al.*? Under what assumptions does this result apply? (2 points)

c) Give the competitive factor for the earliest-deadline-first (EDF) scheduling policy. Also comment on the implications of this result. (2 points)

PROBLEM 7

Consider the following multiprocessor scheduling policies:

- Rate-Monotonic (RM)
- Rate-Monotonic Utilization Separation (RM-US)
- Rate-Monotonic First-Fit (RM-FF)
- Earliest-Deadline-First Utilization Separation (EDF-US)
- Earliest-Deadline-First First-Fit (EDF-FF)

a) Explain the idea behind *utilization separation* (as used in RM-US and EDF-US.) (2 points)

b) Explain the general *first-fit* assignment algorithm (as used in RM-FF and EDF-FF.) (3 points)

c) For each of the listed scheduling policies, there is a guarantee *utilization bound*, which the per-processor utilization of the task set should not exceed in order to guarantee that the tasks will meet their deadlines. Note: For EDF-US and EDF-FF the bound refers to the *synthetic* utilization of the aperiodic tasks.

Now assume that you are designing a real-time system with 10 processors, where the utilization of the task sets being used will never exceed 40% of the total computational capacity of the system. Decide which of the multiprocessor scheduling policies listed above (RM, RM-US, RM-FF, EDF-US, EDF-FF) that will be guaranteed to schedule (using their corresponding utilization bound) any of the task sets that may be used on the system. (5 points)
