Real Time and Embedded Systems/Problem Set #1

Due at the beginning of Lecture 7, Wednesday, 4 February 2004.

1. Consider the real-time program described by the pseudocode below. Names of jobs are **italicized**:

> At 9:00, **start**: have breakfast and go to office;
> At 10:00,
>   If there is class,
>   **teach**;
>   Else,
>   **help** students;
> When *teach* or *help* is done, **eat_lunch**;
> Until 14:00, **sleep**;
> If there is a seminar,
>   If the topic is interesting,
>   **listen**;
>   Else
>   **read**;
> Else
>   **write** in office;
> When seminar is over, **attend** social hour;
> **discuss**;
> **jog**;
> **eat_dinner**;
> **work** a little more
> **end_the_day**;

   a. Draw a task graph to capture the dependencies among jobs.

   b. Use as many precedence graphs as needed to represent all possible paths of the program.

2. A system contains nine nonpreemptable jobs named J_i, for i = 1, 2, …, 9. Their execution times are 3, 2, 2, 2, 4, 4, 4, 4, and 9, respectively, their release times are equal to 0, and their deadlines are 12. J_1 is the immediate predecessor of J_9, and J_4 is the immediate predecessor of J_5, J_6, J_7, and J_8. There are no other precedence constraints. For all jobs, J_i has a higher priority than J_k if i < k.

   a. Draw the precedence graph of the jobs.

   b. Can the jobs meet their deadlines if they are scheduled on three processors? Explain your answer.

   c. Can the jobs meet their deadlines if we make them preemptable and schedule them preemptively. Explain your answer.

   d. Can the jobs meet their deadlines if they are scheduled nonpreemptively on four processors? Explain your answer.

   e. Suppose that due to an improvement of the three processors, the execution time of every job is reduced by 1. Can the jobs now meet their deadlines? Explain your answer.

3. A system contains the following periodic tasks: $T_1 = (5, 1)$, $T_2 = (7, 1, 9)$, $T_3 = (10, 3)$, and $T_4 = (35, 7)$.

   a. If the frame size constraint (5-1) is ignored, what are the possible frame sizes?

   b. Using the largest frame size you have found in part a, draw the network-flow graph of the system.

   c. Find a cyclic schedule by solving the network-flow problem in part b or show that the system cannot be feasibly according to a cyclic schedule of the frame size used in part b.

4. This problem is concerned with the performance and behaviour of rate-monotonic and earliest-deadline-first algorithms.

   a. Construct the initial segments in the time interval (0, 750) of a rate-monotonic schedule and an earliest-deadline-first schedule of the periodic tasks (100, 20), (150, 50), and (250, 100) whose total utilization is 0.93.

   b. Construct the initial segments in the time interval (0, 750) of a rate-monotonic schedule and an earliest-deadline-first schedule of the periodic tasks (100, 20), (150, 50), and (250, 120) whose total utilization is 1.1.