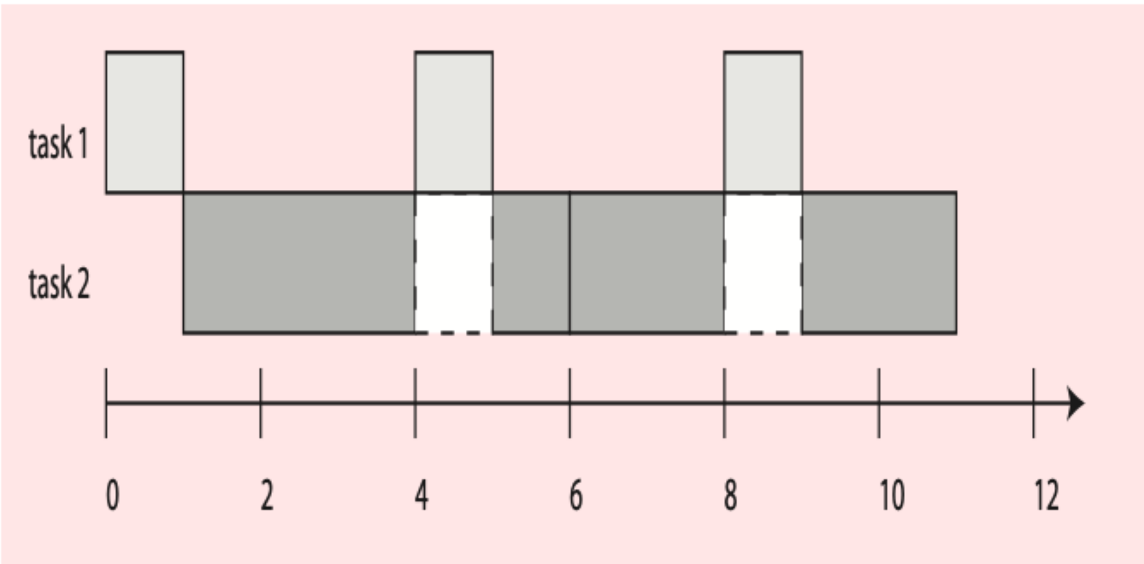
Exercise - Scheduling

Ex 1

This problem studies fixed-priority scheduling. Consider two tasks to be executed periodically on a single processor, where task 1 has period p1 = 4 and task 2 has period p2 = 6.

a) Let the execution time of task 1 be e1 = 1. Find the maximum value for the execution time e2 of task 2 such that the RM schedule is feasible.

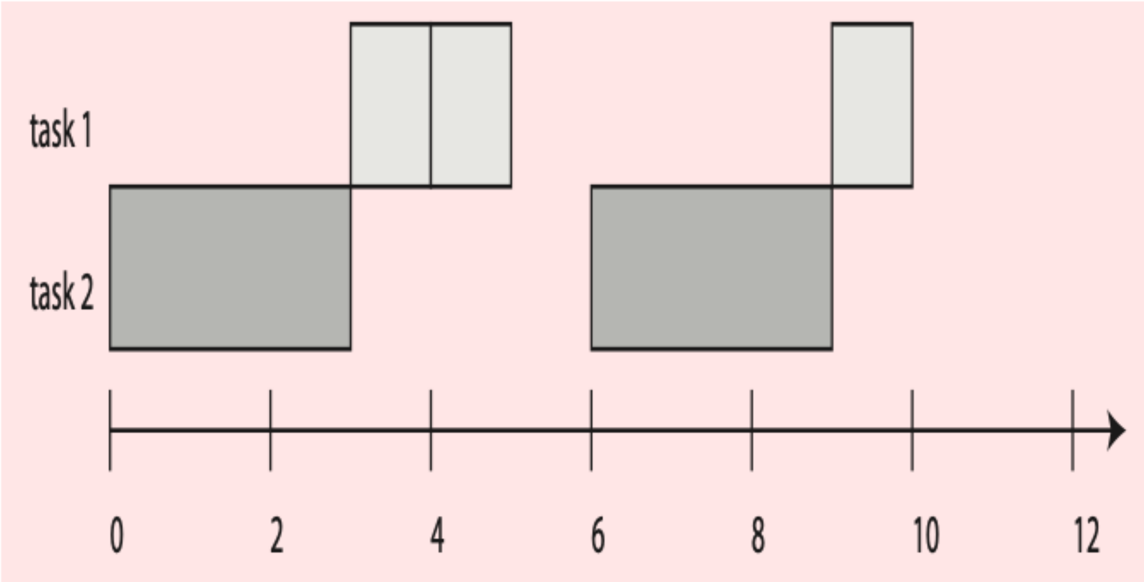
Sol: The largest execution time for task2 is e2 = 4. The following figure shows the resulting schedule:



The schedule repeats every 12 time units.

b) Again let the execution time of task 1 be e1 = 1. Let non-RMS be a fixed- priority schedule that is not an RM schedule. Find the maximum value for the execution time e2 of task 2 such that non-RMS is feasible.

Sol: The largest execution time for task2 is e2 = 3. The following figure shows the resulting schedule:



The schedule repeats every 12 time units.

c) For both your solutions to (a) and (b) above, find the processor utilization. Which is better?

Sol: From the figures above, we see that the RM schedule results in the machine being idle for 1 out of 12 time units, so the utilization is 11/12. The non-RM schedule results in the machine being idle for 3 out of 12 time units, so the utilization is 9/12 or 3/4. The RM schedule is better. i.e.:

RM: 1/4 + 4/6 = 3/12 + 8/12 = 11/12

Non-RM: 1/4 + 3/6 = 3/12 + 6/12 = 9/12 = 3 /4 (RM scheduller is better)

d) For RM scheduling, are there any values for e1 and e2 that yield 100% utiliza- tion? If so, give an example.

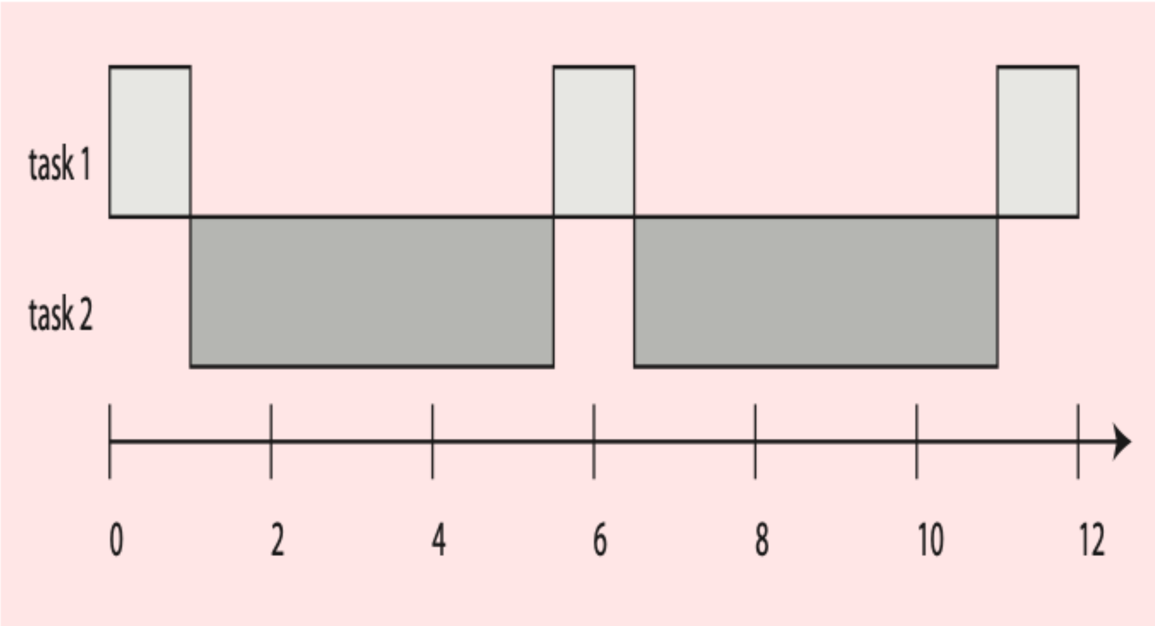
Sol: Yes. For example, e1 = 4 and e2 = 0.

Ex 2

This problem studies dynamic-priority scheduling. Consider two tasks to be executed periodically on a single processor, where task 1 has period p1 = 4 and task 2 has period p2 = 6. Let the deadlines for each invocation of the tasks be the end of their period. That is, the first invocation of task 1 has deadline 4, the second invocation of task 1 has deadline 8, and so on.

a) Let the execution time of task 1 be e1 = 1. Find the maximum value for the execution time e2 of task 2 such that EDF is feasible.

Sol: The largest execution time for task 2 is e2 = 4.5. The following figure shows the resulting schedule:



The schedule repeats every 12 time units.

b) For the value of e2 that you found in part (a), compare the EDF schedule against the RM schedule from Exercise 1 (a). Which schedule has less preemption? Which schedule has better utilization?

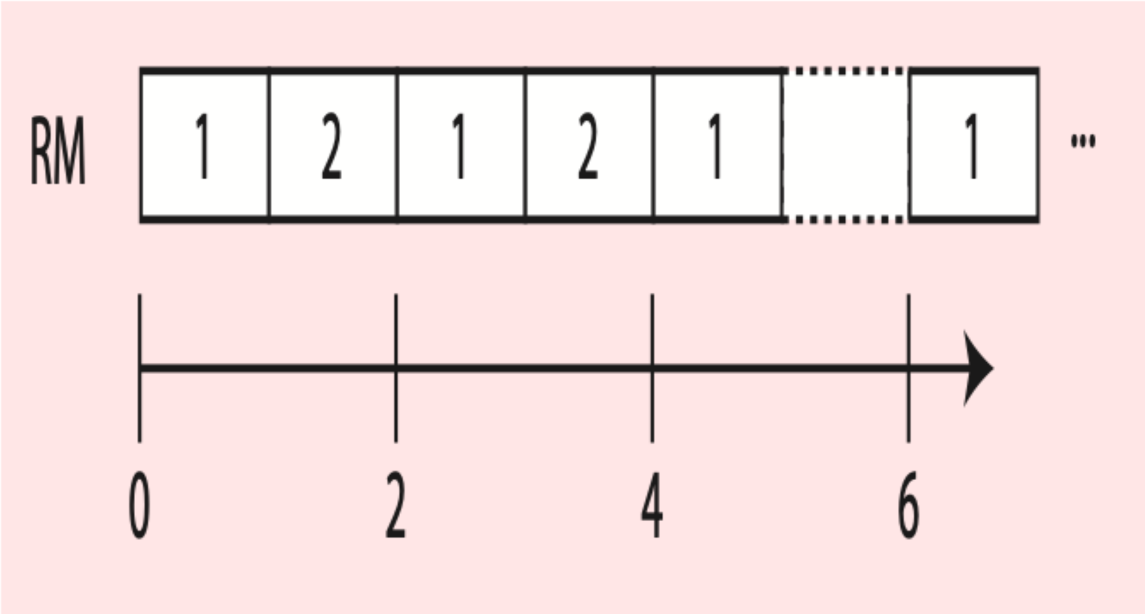
Sol: Comparing the schedule in (a) with the schedule in Exercise 1(a), we see that EDF has no preemption at all, while RM performs two preemptions every 12 time units. Moreover, EDF has 100% utilization, whereas RM has less.

Ex 3

This problem compares RM and EDF schedules. Consider two tasks with periods *p1 = 2* and *p2 = 3* and execution times *e1 = e2 = 1*. Assume that the deadline for each execution is the end of the period.

a) Give the RM schedule for this task set and find the processor utilization. How does this utilization compare to the Liu and Layland utilization bound given by $\mu \le n(2\text{^}(1/n)-1)$?

Sol: The RM schedule is shown below:



The utilization is given by

$U = 1 - 1/6 \approx 83.3\%$

The utilization bound if $n = 2$ is $n(2^{(1/n)} - 1) \approx 0.828$. Thus, utilization is larger than the utilization bound, so we have no assurance that the RM schedule is feasible.

b) Show that any increase in *e1* or *e2* makes the *RM* schedule infeasible.

Sol: In the first three time units, the RM schedule must execute task1 twice, because under the RM principle, it has highest priority and it has become enabled twice in this time period. With e1 = 1, this leaves exactly one time unit to execute task 2 in its first period. Thus, any increase in e2 will result in task 2 missing its deadline at time 3. Any increase in e1 will leave less than one time unit for task 2 in its first period, resulting again in a missed deadline.

c) If you hold *e1 = e2 = 1* and *p2 = 3* constant, is it possible to reduce *p1* below 2 and still get a feasible schedule? By how much?

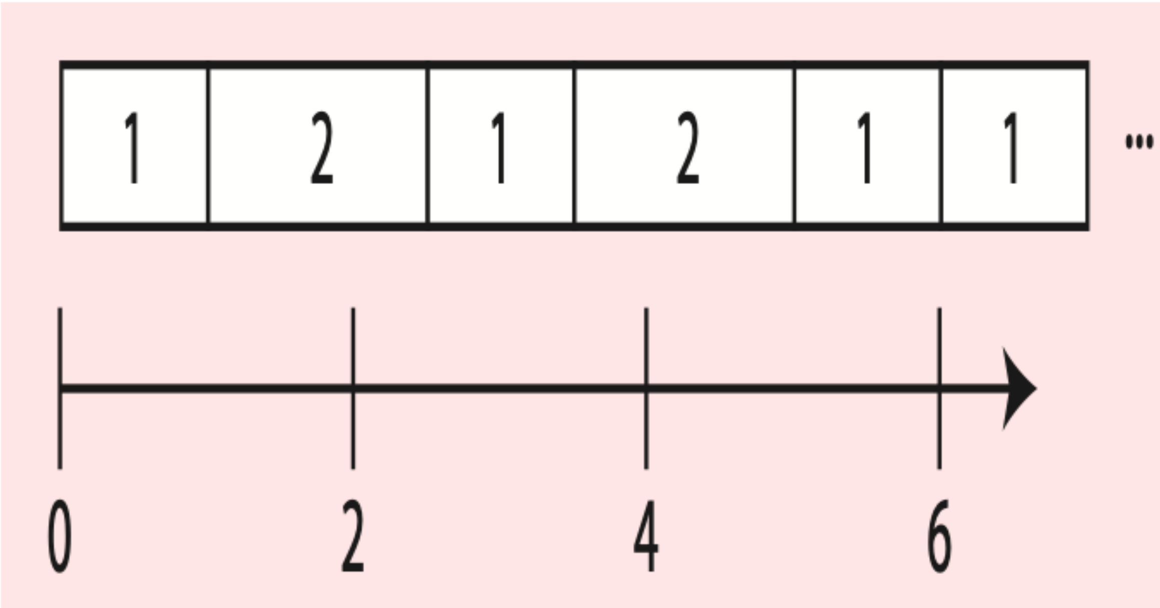
Sol: Holding e1, e2, and p2 constant, we can reduce p1 to 1.5 and still get a feasible schedule.

d) If you hold *e1 = e2 = 1* and *p1 = 2* constant, is it possible to reduce *p2* below 3 and still get a feasible schedule? By how much?

Sol: Holding e1, e2, and p1 constant, we can reduce p2 to 2 and still get a feasible schedule. In both cases ((c) and (d)), no further reduction is possible because at this point we have 100% utilization.

e) Increase the execution time of task 2 to be *e2 = 1.5*, and give an *EDF* schedule. Is it feasible? What is the processor utilization?

Sol: The EDF schedule is:



The schedule is feasible and the utilization is 100%.