Q6-2 Name an embedded system that requires both periodic and aperiodic computation.

Q6-3 An audio system processes samples at a rate of 44.1 kHz. At what rate could we sample the system's front panel to both simplify analysis of the system schedule and provide adequate response to the user's front panel requests?

Q6-4 Draw a UML class diagram for a process in an operating system. The process class should include the necessary attributes and behaviors required of a typical process.

Q6-5 Draw a task graph in which P1 and P2 each process separate inputs and then pass their results onto P3 for further processing.

Q6-6 Compute the utilization for these task sets:

a. P1: period = 1 s, execution time = 10 ms; P2: period = 100 ms, execution time = 10 ms

b. P1: period = 100 ms, execution time = 25 ms; P2: period = 80 ms, execution time = 15 ms; P3: period = 40 ms, execution time = 5 ms.

c. P1: period = 10 ms, execution time = 1 ms; P2: period = 1 ms, execution time = 0.2 ms; P3: period = 0.2 ms, execution time = 0.05 ms.

Q6-7 What factors provide a lower bound on the period at which the system timer interrupts for preemptive context switching?

Q6-8 What factors provide an upper bound on the period at which the system timer interrupts for preemptive context switching?

Q6-9 What is the distinction between the ready and waiting states of process scheduling?

Q6-10 A set of processes changes state as shown over the interval [0, 1 ms]. P1 has the highest priority and P3 has the lowest priority. Draw a UML sequence diagram showing the state of all the processes during this interval.

t	Process states
0	P1 = waiting, P2 = waiting, P3 = executing
0.1	P1 = ready
0.15	P2 = ready
0.2	P1 = waiting
0.3	P1 = ready, P3 = ready
0.4	P1 = waiting
0.5	P2 = waiting
0.6	P3 = waiting
0.8	P2 = ready, P3 = ready
0.9	P2 = waiting

Q6-11 Provide examples of

a. blocking interprocess communication;

b. nonblocking interprocess communication.

Q6-12 For the following periodic processes, what is the shortest interval we must examine to see all combinations of deadlines?

a.

Process	Deadline
P1	2
P2	5
P3	10

b.

Process	Deadline
P1	2
P2	4
P3	5
P4	10

c.

Process	Deadline
P1	3
P2	4
P3	5
P4	6
P5	10

Q6-13 Consider the following system of periodic processes executing on a single CPU:

Process	Execution time	Deadline
P1	4	200
P2	1	10
P3	2	40
P4	6	50

Can we add another instance of P1 to the system and meet all the deadlines using RMS?

Q6-14 Given the following set of periodic processes running on a single CPU (P1 has highest priority), what is the maximum execution time x of P3 for which all the processes will be schedulable using EDF?

Process	Execution time	Deadline
P1	1	10
P2	3	25
P3	X	50
P4	10	100

Q6-15 A set of periodic processes is scheduled using RMS; P1 has the highest priority. For the process execution times and periods shown below, show the state of the processes at the critical instant for each of these processes.

a. P1
b. P2
c. P3

Process	Time	Deadline
P1	1	4
P2	1	5
P3	1	10

Q6-16 For the given periodic process execution times and periods (P1 has the highest priority), show how much CPU time of higher-priority processes will be required during one period of each of the following processes:

a. P1
b. P2
c. P3
d. P4

Process	Time	Deadline
P1	1	5
P2	2	10
P3	2	25
P5	5	50

Q6-17 For the periodic processes shown below:

a. Schedule the processes using an RMS policy.
b. Schedule the processes using an EDF policy.

In each case, compute the schedule for an interval equal to the least-common multiple of the periods of the processes. P1 has the highest priority and time starts at $t = 0$.

Process	Time	Deadline
P1	1	3
P2	1	4
P3	1	12

Q6-18 For the periodic processes shown below:

a. Schedule the processes using an RMS policy.
b. Schedule the processes using an EDF policy.

In each case, compute the schedule for an interval equal to the least-common multiple of the periods of the processes. P1 has the highest priority and time starts at $t = 0$.

Process	Time	Deadline
P1	1	3
P2	1	4
P3	2	6

Q6-19 For the periodic processes shown below:

a. Schedule the processes using an RMS policy.
b. Schedule the processes using an EDF policy.

In each case, compute the schedule for an interval equal to the least-common multiple of the periods of the processes. P1 has the highest priority and time starts at $t = 0$.

Process	Time	Deadline
P1	1	2
P2	1	3
P3	2	10

Q6-20 For the given set of periodic processes, all of which share the same deadline of 12:

a. Schedule the processes for the given arrival times using standard rate-monotonic scheduling (no data dependencies).

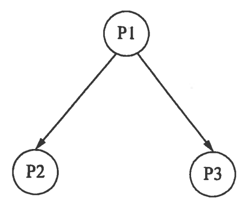
b. Schedule the processes taking advantage of the data dependencies. By how much is the CPU utilization reduced?

Process	Execution time
P1	2
P2	1
P3	2

Q6-21 For the periodic processes given below, find a valid schedule

a. using standard RMS;

b. adding one unit of overhead for each context switch.



Process	Time	Deadline
P1	2	30
P2	5	40
P3	7	120
P4	5	60
P5	1	15

Q6-22 For the periodic processes and deadlines given below:

a. Schedule the processes using RMS.

b. Schedule using EDF and compare the number of context switches required for EDF and RMS

Process	Time	Deadline
P1	1	5
P2	1	10
P3	2	20
P4	10	50
P5	7	100

Q6-23 If you wanted to reduce the cache conflicts between the most computationally intensive parts of two processes, what are two ways that you could control the locations of the processes' cache footprints?

Q6-24 A system has two processes P1 and P2 with P1 having higher priority. They share an I/O device ADC. If P2 acquires the ADC from the RTOS and P1 becomes ready, how does the RTOS schedule the processes using priority inheritance?

Q6-25 Explain the roles of interrupt service routines and interrupt service handlers in interrupt handling.

Q6-26 Briefly explain the dual-kernel approach to RTOS design.

Q6-27 What are the kernel-level units of execution in WinCE?

Q6-28 How would you use the ADPCM method to encode an unvarying (DC) signal with the coding alphabet $\{-3, -2, -1, 1, 2, 3\}$?

Lab exercises

L6-1 Using your favorite operating system, write code to spawn a process that writes "Hello, world" to the screen or flashes an LED, depending on your available output devices.

L6-2 Build a small serial port device that lights LEDs based on the last character written to the serial port. Create a process that will light LEDs based on keyboard input.

L6-3 Write a driver for an I/O device.

L6-4 Write context switch code for your favorite CPU.

L6-5 Measure context switching overhead on an operating system.