Review Problems

[1] A hard real-time system runs three tasks using the cyclic executive method. Tasks have periods of 10 ms, 12 ms, and 15 ms and execution times per period of 2, 2, and 5 ms respectively. The deadline of each task is equal to its period.

a) If the deadline of each task is equal to its period, find a suitable frame value then construct an execution schedule for the tasks if possible.
b) If the deadline of third tasks is 10 ms (with period of 15 ms) is the frame value you obtained in part (a) still suitable? Is it possible to find an execution schedule?

---

[2] A real-time system runs three tasks using the cyclic executive method. Tasks have periods of 30 ms, 18 ms, and 20 ms and execution times per period of 6, 3, and 5 ms respectively. The deadline of each task is equal to its period.

a) Show that 10 ms is a suitable minor cycle (frame) value.
b) Find an execution schedule for the above frame value (if possible).
c) In addition to the above periodic tasks system is required to run an aperiodic task released at time t=200 ms with an execution time of 4 ms. How would the system handle this task and what is the earliest time its execution can end?

---

[3] A hard real-time system is required to run three periodic tasks with periods of 50, 20, and 70 ms, and execution times per period of 10, 5, and 25 ms respectively. The deadline of each task is equal to its period.

a) Are the above three tasks RM schedulable? Explain your answer.
b) By which amount can the period of the third task be reduced while the three tasks remain EDF schedulable?
c) Assume that fixed priorities are used, but the first task is a soft real-time task for which the execution time can increase to 16 ms with low probability. How would you assign the priorities in this case? Explain your answer.

---

[4] A hard real-time system is required to run three independent periodic tasks with periods of 70, 40, and 20 ms, and execution times of 12, 13, and 8 ms per period respectively. The deadline of each task is equal to its period.

a) Are the above tasks schedulable using the RM algorithm? Explain your answer.
b) Repeat part (a) for the EDF scheduling algorithm.
c) In addition to the above three tasks the system is required to run five tasks with CPU utilizations of 0.71, 0.47, 0.21, 0.55, and 0.16. Use the first-fit decreasing assignment algorithm to find the minimum number of required processors, assuming that all processors use EDF scheduling.

---

[5] A system is required to run three hard real-time tasks with periods of 15, 20, and 30 ms, and execution times of 4, 5, and 10 ms per period respectively.

a) If the deadline of each task is equal to its period, use response-time analysis to find whether tasks will always meet their deadlines with Rate-Monotonic priority assignment.
b) Repeat part (a) if first and third tasks use a shared resource exclusively for 4 ms each. Assume that priority inheritance is applied.
c) Repeat part (a) if deadlines of tasks are 15, 24, and 20 ms respectively.
d) Repeat *part (c)* assuming Deadline-Monotonic fixed priority assignment.

---

[6] Consider the following three processes that run concurrently at unknown speeds. Assume that initially m=3 and n=0.

Process 1	Process 2	Process 3
….	….	….
while (1)	while (1)	while (1)
{ down (m);	{ down (n);	{ down (n);
printf ("A ");	printf ("B ");	printf ("D ");}
up (n);}	printf ("C ");	
	up (n);}	

If the three processes run until they are all blocked:

a) How many A's are displayed?
b) How many D's are displayed?
c) What is the minimum number of B's that might be displayed?
d) Is A B C B A D C A B C D D a possible output sequence? Explain.

---

[7] A thread is required to be blocked until *n* other threads end. These other threads can run in any order. Show how this can be done using the semaphores method.

---

[8] Consider the following three processes:

Process P1	Process P2	Process P3
…..	…..	…..
for (i=0; i < 50; i++)	for (j=0; j < 50; j++)	for (k=0; k < 50; k++)
A[i]= fn1(i);	B[j]= fn2(j);	C[k]= A[k]+B[k];
…..	…..	…..

a) Show how to use semaphores such that P3 does not modify an element of C[ ] until the corresponding elements of A[ ] and B[ ] are modified by P1 and P2.
b) Can the above requirement be satisfied using only one semaphore? Explain?
c) Show how to use semaphores such that no elements of C[ ] are modified until the whole arrays A[ ] and B[ ] are modified.