CSE4300 Homework 2 (Due on April 9th, 2019)

Question 1 (10 points). Many CPU-scheduling algorithms are parameterized. For example, the RR algorithm requires a parameter to indicate the time slice. Multilevel feedback queues require parameters to define the number of queues, the scheduling algorithm for each queue, the criteria used to move processes between queues, and so on.

These algorithms are thus really sets of algorithms (for example, the set of RR algorithms for all time slices, and so on). One set of algorithms may include another (for example, the FCFS algorithm is the RR algorithm with an infinite time quantum). What (if any) relation holds between the following pairs of algorithm sets?

a. Priority and SJF
b. Multilevel feedback queues and FCFS
c. Priority and FCFS
d. RR and SJF

Question 2 (25 points). Consider the following set of processes, with the length of the CPU burst given in milliseconds:

Process	Burst Time	Priority
$P1$	2	2
$P2$	1	1
$P3$	8	4
$P4$	4	2
$P5$	5	3

The processes are assumed to have arrived in the order $P1$, $P2$, $P3$, $P4$, $P5$, all at time 0.

a. Draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, nonpreemptive priority (a larger priority number implies a higher priority), and RR (quantum = 2).

b. What is the turnaround time of each process for each of the scheduling algorithms in part a?

c. What is the waiting time of each process for each of these scheduling algorithms?

d. Which of the algorithms results in the minimum average waiting time (over all processes)?

Question 3 (25 points). Given a synchronous task set of three periodic tasks: T1 = (1, 6, 6), T2 = (2, 8, 3) and T3 = (4, 12, 8), please construct the schedules for the task set from time 0 to time 24 under:

1) Preemptive RM, DM, EDF

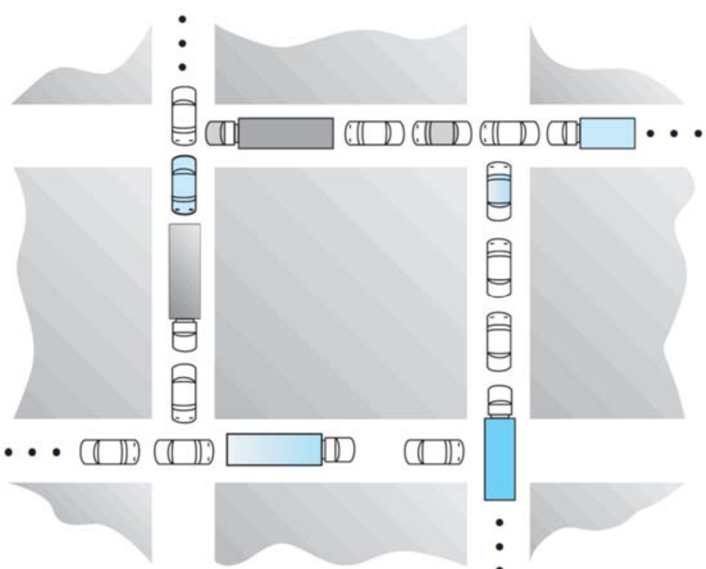
2) Non-Preemptive RM, DM, EDF

3) Calculate the average response time of the constructed six schedules

Question 4 (20 points). Consider the traffic deadlock depicted in the following figure.

a. Show that the four necessary conditions for deadlock hold in this example.

b. State a simple rule for avoiding deadlocks in this system.



Question 5 (20 points): A system has four processes and five allocable resources. The current allocation and request matrix are as follows:

	Current Allocation Matrix
Process A	1 0 2 0 1
Process B	2 0 1 1 0
Process C	1 0 0 1 1
Process D	1 1 1 0 1

	Request Matrix
Process A	2 0 2 0 0
Process B	2 0 0 3 0
Process C	1 1 0 0 0
Process D	0 0 0 0 1

Available resources: 1 1 1 x 1

For what value of x this will be a safe state?

Show a sequence of scheduling (if exists) that will lead to successful completion of the tasks.

The request matrix specifies how many of each type of resources are still needed by each process, in addition to what it currently has.