BIRZEIT UNIVERSITY
Electrical and Computer Engineering Department

ENCS 339 Operating Systems

First Semester, 2018-2019 HW#3 Due Date: Sunday October 28th 2018

Deadlock Prevention:

Deadlock or starvation: **3 points: 1 each**

1- 4 cars at a crossroad with a stop sign each: the car to right moves first: • **Deadlck** • **Starvation**

2- Many cars at a cross road: one road is MAIN another is auxiliary: the car on the MAIN moves first: • **Deadlck** • **Starvation**

3- We have 4 processes in a system. It is possible to have a deadlock between 2 processes only and the other 2 can still be working fine. • **True** • **False**

2 Deadlock Avoidance

Consider the following snapshot of a system:

1- (2points) According to this snapshot, the system is in a safe state. Show the order in which the processes can finish (just ass index#):

P0 → P3 → P1 → P2 → P4
OR P3 → P0 → P1 → P2 → P4

(2points) Is this the only possible order? • **Yes** • **No**

2- (2 point) How many instances of each resources the system has:

A	B	C	D
3	14	12	12

3- (2points) If a request from process P_0 arrives for (0,3,0,0), can the request be granted immediately? • **Yes** • **No**

P0 Alloc:0312→ **P0 → P3 → P1 → P2 → P4**

Av: 1220

4- (2points) If instead request from process P_2 arrives for (1,0,1,0), can the request be granted immediately? • **Yes** • **No**

Exceeds limit(Max)

5- (2points) If instead request from process P_4 arrives for (0,3,1,0), can the request be granted immediately? • **Yes** • **No**

P4 Alloc:0324→ **P0 cannot start: P3 → P0 → P1 → P2 → P4**

Av: 1210 1842 1854 2854 3 11 10 8 3 14 12 12

	Allocation				Max				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	0	0	1	2	0	3	1	2	1	5	2	0
P_1	1	0	0	0	1	7	5	0				
P_2	1	3	5	4	2	3	5	6				
P_3	0	6	3	2	0	6	5	2				
P_4	0	0	1	4	0	6	5	6				

Good Luck