

Chapter 8 - Deadlock

- System consists out of processes sharing limited resources
- Processes utilize resources in the following manner:
 1. request a resource
 2. use the resource
 3. release the resource

1

Chapter 8 - Conditions

1. Mutual exclusion
2. Hold and wait
3. No preemption
4. Circular Wait.

2

Chapter 8 - Handling Deadlocks

- Three possibilities:
 1. System uses a protocol to avoid deadlock
 2. System allows deadlock, but can recover
 3. System ignores the problem
- Difference between **avoidance** and **prevention**

3

Chapter 8 - Prevention

- System ensures that at least one of the conditions necessary for deadlock does not hold
- Mutual Exclusion: Not all resources can be shared
- Hold and Wait: Processes can only allocate resources if they do not hold any other. Disadvantages?
- Preemption
- Circular Wait: Resources are prioritized by ordering

4

Chapter 8 - Avoidance

- Guarantee that system is always in a safe state
- Resource allocation graphs
- Banker's Algorithm

5

Example: Safe and unsafe states

P_0 (10)	P_1 (4)	P_2 (9)	Available
5	2	2	3
-	2	-	1
-	(4)	-	5
5	-	-	0
(10)	-	-	10
-	-	7	2

6

Chapter 8 - Deadlock Detection

- System requirements:
 1. Algorithm to recognize deadlock
 2. Algorithm to recover from deadlock
- Single instance of a resource: wait-for graph
- Multiple instances of a resource
- When should the aforementioned algorithms be used

7

Chapter 8 - Deadlock Recovery

- Terminate processes
- How are processes chosen?
- Preemption of resources
 1. Which process, which resource
 2. Rollback
 3. Starvation
- Combination of methods

8