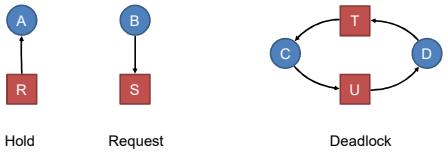
CSC 452: Deadlocks

Dr. Jonathan Misurda
jmisurda@cs.arizona.edu
http://www.u.arizona.edu/~jmisurda

Modeling Deadlocks



Hold Request Deadlock

"A set of processes is deadlocked if each process in the set is waiting for an event that only another process in the set can cause."

Dealing with Deadlocks

1. Ignore
2. Detect and recover
3. Avoid
4. Prevent

4 Conditions for Deadlock

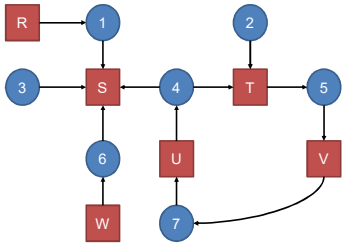
1. Mutual exclusion
   - Resource can only be held by one process at a time
2. Hold and wait
   - Process gains one resource, holds it, then attempts to gain another, waiting if failed
3. No preemption
   - Resource cannot be forcibly taken away
4. Circular wait
   - Process A is waiting for a resource held by Process B which is waiting for a resource held by Process A …

Ostrich Algorithm

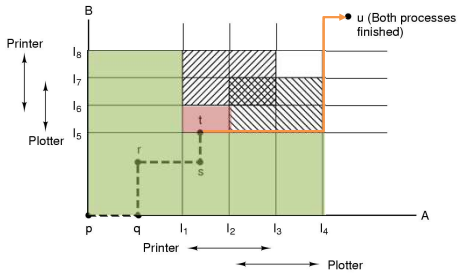
Do nothing – pretend like it didn't happen

Deadlock Detection

Process	Holds	Wants
1	R	S
2		T
3		S
4	U	S,T
5	T	V
6	W	S
7	V	U



Resource Trajectories



Deadlock Detection Algorithm

```
For each node N in the graph {
  Set L = empty list
  unmark all arcs
  Traverse (N,L)
}

If no deadlock reported by now, there isn't any

define Traverse (C,L)  {
  If C in L, report deadlock!
  Add C to L
  For each unmarked arc from C {
    Mark the arc
    Set A = arc destination
    /* NOTE: L is a local variable */
    Traverse (A,L)
  }
}
```

Safe State

There exists a schedule that will not lead to deadlock

Deadlock Avoidance

Banker's Algorithm

	Has	Max
A	0	6
B	0	5
C	0	4
D	0	7

Free: 10

Safe State

	Has	Max
A	1	6
B	1	5
C	2	4
D	4	7

Free: 2

Safe State

	Has	Max
A	1	6
B	2	5
C	2	4
D	4	7

Free: 1

Unsafe State

Banker's with Multiple Resources

Process	Tape drives	Plotters	Scanners	CD ROMs
A	3	0	1	1
B	0	1	0	0
C	1	1	1	0
D	1	1	0	1
E	0	0	0	0

Resources assigned

Process	Tape drives	Plotters	Scanners	CD ROMs
A	1	1	0	0
B	0	1	1	2
C	3	1	0	0
D	0	0	1	0
E	2	1	1	0

Resources still needed

E = (6342)
P = (5322)
A = (1020)

Deadlock Prevention

- Attack Mutual Exclusion
- Attack Hold and Wait
- Attack No Preemption
- Attack Circular Wait