The Deadlock Problem

- A set of blocked processes each holding a resource and waiting to acquire a resource held by another process in the set.
- Example
  - semaphores $A$ and $B$, initialized to 1
    
    \[
    \begin{align*}
    &P_0 & P_1 \\
    &\text{wait (A)} & \text{wait (B)} \\
    &\text{wait (B)} & \text{wait (A)}
    \end{align*}
    \]
- Example
  - System has 2 tape drives.
    - $P_1$ and $P_2$ each hold one tape drive and each needs another one.

Bridge Crossing Example

- Traffic only in one direction.
- Each section of a bridge can be viewed as a resource.
- If a deadlock occurs, it can be resolved if one car backs up (preempt resources and rollback).
- Several cars may have to be backed up if a deadlock occurs.
- Starvation is possible.

Deadlock with more than 2 resources

Deadlock Characterization

Deadlock can arise if four conditions hold simultaneously.

- **Mutual exclusion**: only one process at a time can use a resource.
- **Hold and wait**: a process holding at least one resource is waiting to acquire additional resources held by other processes.
- **No preemption**: a resource can be released only voluntarily by the process holding it, after that process has completed its task.
- **Circular wait**: there exists a set $\{P_0, P_1, \ldots, P_n\}$ of waiting processes such that $P_0$ is waiting for a resource that is held by $P_1$, $P_i$ is waiting for a resource that is held by $P_{i+1}$, and $P_n$ is waiting for a resource that is held by $P_0$. 