Lecture 8. Deadlocks

(1) What is a deadlock?

Got stuck, no progress due to resource held.
Stuck due to coordination issue. ⇒ Dead state in system FSM

E.g. A B

| wait (M1) | wait (M2) |
| wait (M2) | wait (M1) |
| signal (M2) | signal (M1) |
| signal (M1) | signal (M2) |

Conditions where deadlock happens:
- Mutual exclusion
- Hold & wait
- No preemption
- Circuit wait

One way to avoid deadlocks in practice: Lock ranking
- Assign rank (priority) to each resource,
- Acquire in increasing rank, release in decreasing rank.
- If a process blocks on one resource, it can't get a resource with higher rank. (Each resource has unique rank)

<table>
<thead>
<tr>
<th>Produce item</th>
<th>consumer</th>
</tr>
</thead>
<tbody>
<tr>
<td>* produce item</td>
<td>* down (full)</td>
</tr>
<tr>
<td>* down (mutex)</td>
<td>* solve</td>
</tr>
<tr>
<td>* put items in queue</td>
<td>* up (mutex)</td>
</tr>
<tr>
<td>* up (mutex)</td>
<td>* up (empty)</td>
</tr>
<tr>
<td>* up (full)</td>
<td>consider down = wait</td>
</tr>
<tr>
<td></td>
<td>up = signal</td>
</tr>
<tr>
<td></td>
<td>consider mutex full</td>
</tr>
<tr>
<td></td>
<td>empty</td>
</tr>
<tr>
<td></td>
<td>as resources</td>
</tr>
</tbody>
</table>
P(m) \rightarrow E
\uparrow
C

\begin{align*}
\text{States:} & \quad \text{empty} = 0, \\
& \quad \text{full, empty} > 0, \\
& \quad \text{full} = 0.
\end{align*}

(2) priority inversion
the case we need it \( P_1 > P_2 > P_3 \).
resource held by \( P_3 \),
requested by \( P_1 \),
P_2 is running.

P_1 \quad \xrightarrow{\text{done and request } R}
\quad \text{running}
P_2 \quad \text{holding } R.
P_3 \quad \text{running}

But \( P_2 \) is running, \( P_3 \) doesn't get a chance to release \( R \).
This may block \( P_1 \) for long.

(3) live blocks & starvation
\begin{itemize}
\item example: buffer reassembly.
\item have multiple packets.
\end{itemize}

While true send fragments
\begin{itemize}
\item wait for Ack
\item done
\end{itemize}

\begin{itemize}
\item sends out signal packet
\item we don't want fragments from different packets to mixed up.
\end{itemize}