Rule 1:

All tasks break down into any two fundamental classes:

1. **Synchronous**
   - Non-blocking
   - FAST

2. **Asynchronous**

**DPP - Dining Philosophers Problem**

**Deadlock**

2 Fingers to EAT

**Eating**

1987/77

What is an event?

Event = detachable edge

Event:
Corollary to Rule #1
- Keep end detector and service ready as short as possible.
- Make the most blocking.

Complete Program Structure
- Initialize all input/output
  \texttt{while(1)}
  \begin{itemize}
  \item Test for events - read Robin
  \item Service then exit
  \end{itemize}

State Machine
- Description of an abstract machine
- At any point in time, all state paths can only be in one of a fixed number of states.
- Next state in a program depends solely on the current state and the input/output.
- Idealized, real-time transition from one state to the next.

---

Bomb
\begin{itemize}
\item Total damage
\item Kills everyone
\item Doomsday device
\end{itemize}
Questions?