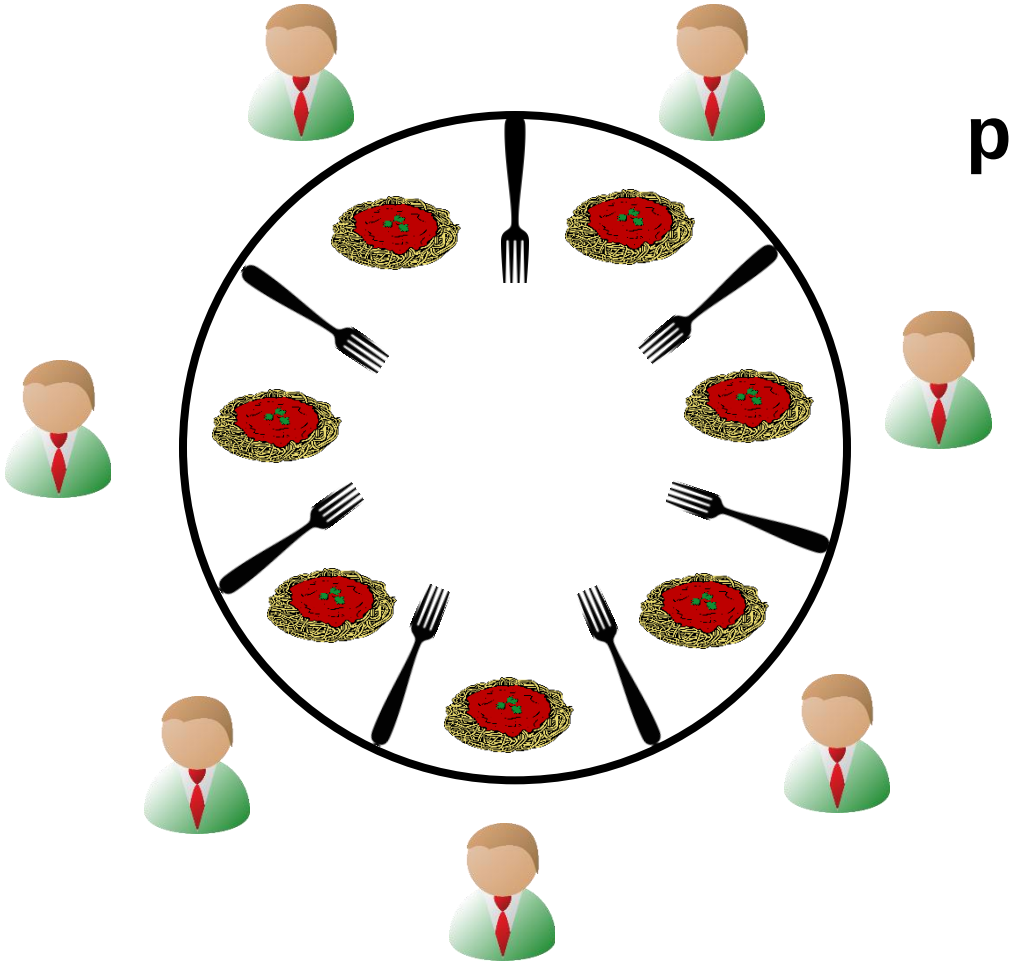
Dinning Philosophers Problem

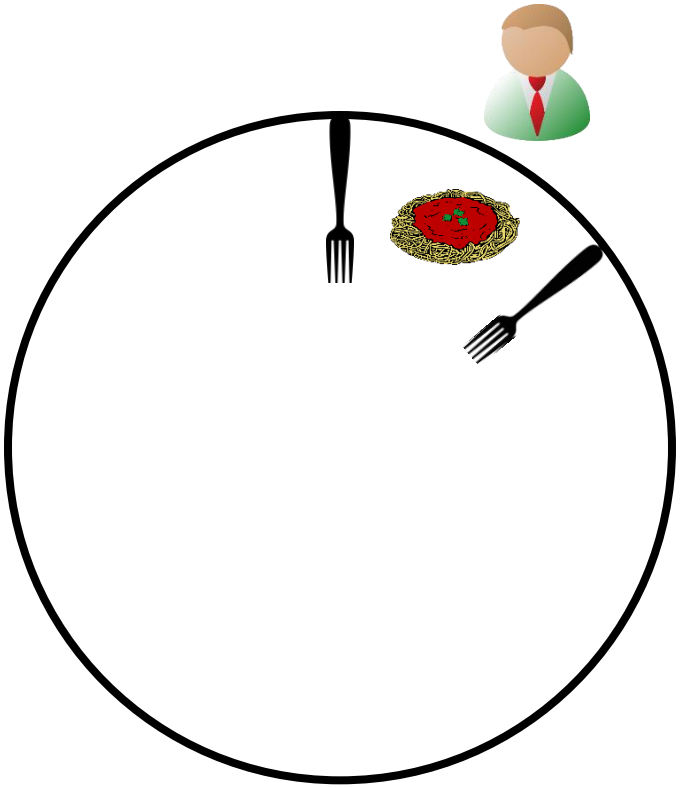
The forks are locks



The philosophers are threads

A philosopher

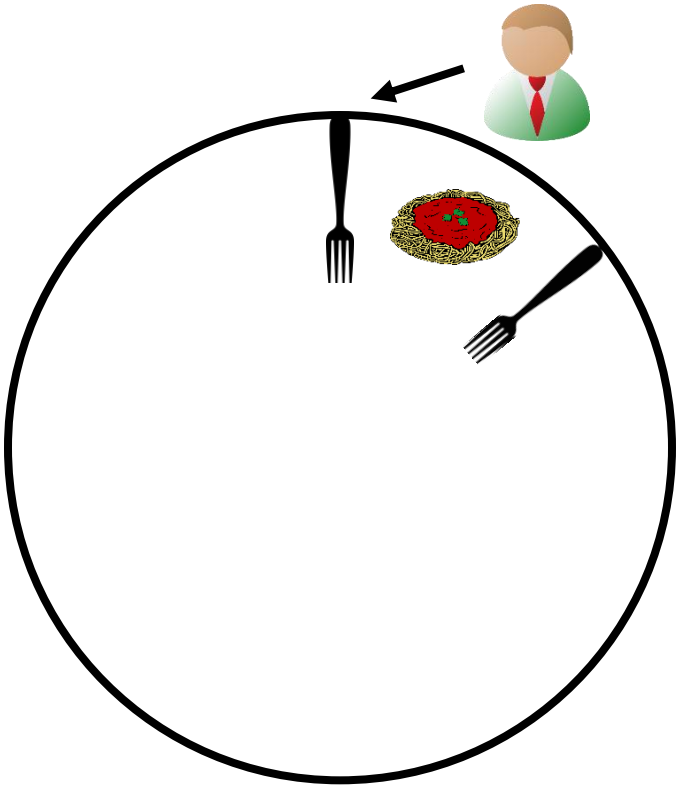
The forks are locks



A philosopher wants to eat.
Eat is a task that required both forks (**locks**).
Eat could simply mean displaying a message.

A philosopher

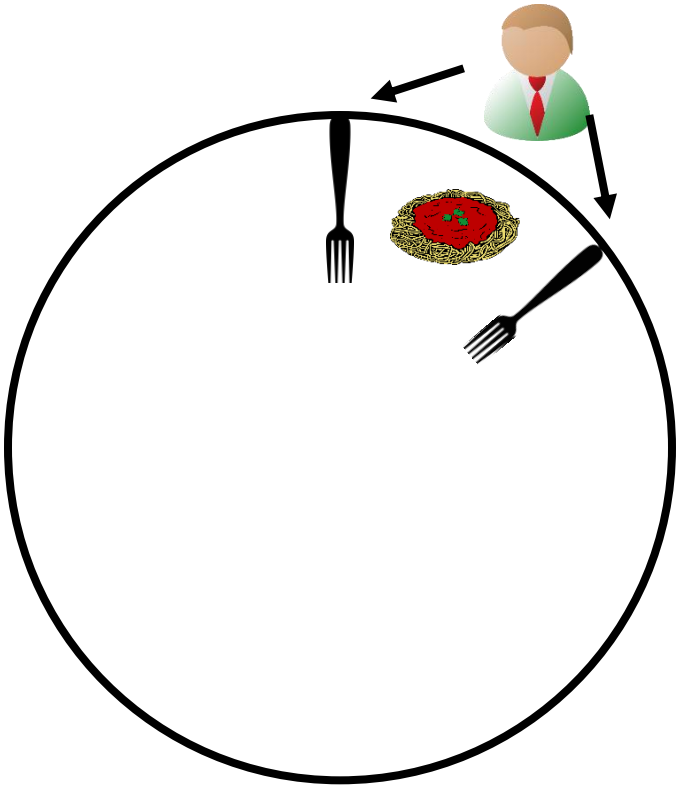
The forks are locks



Take right fork (lock)

A philosopher

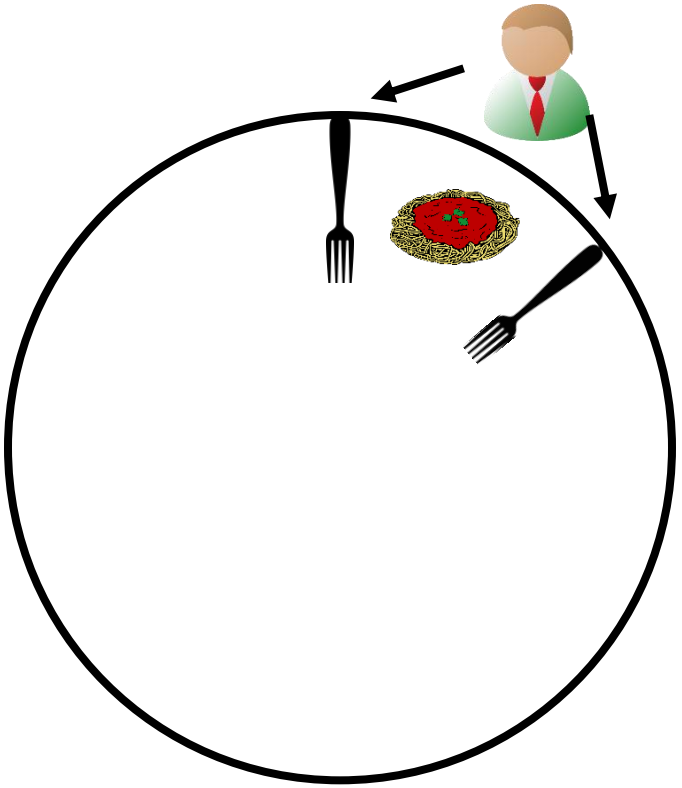
The forks are locks



While holding right fork (lock) Take the left fork (lock)

A philosopher

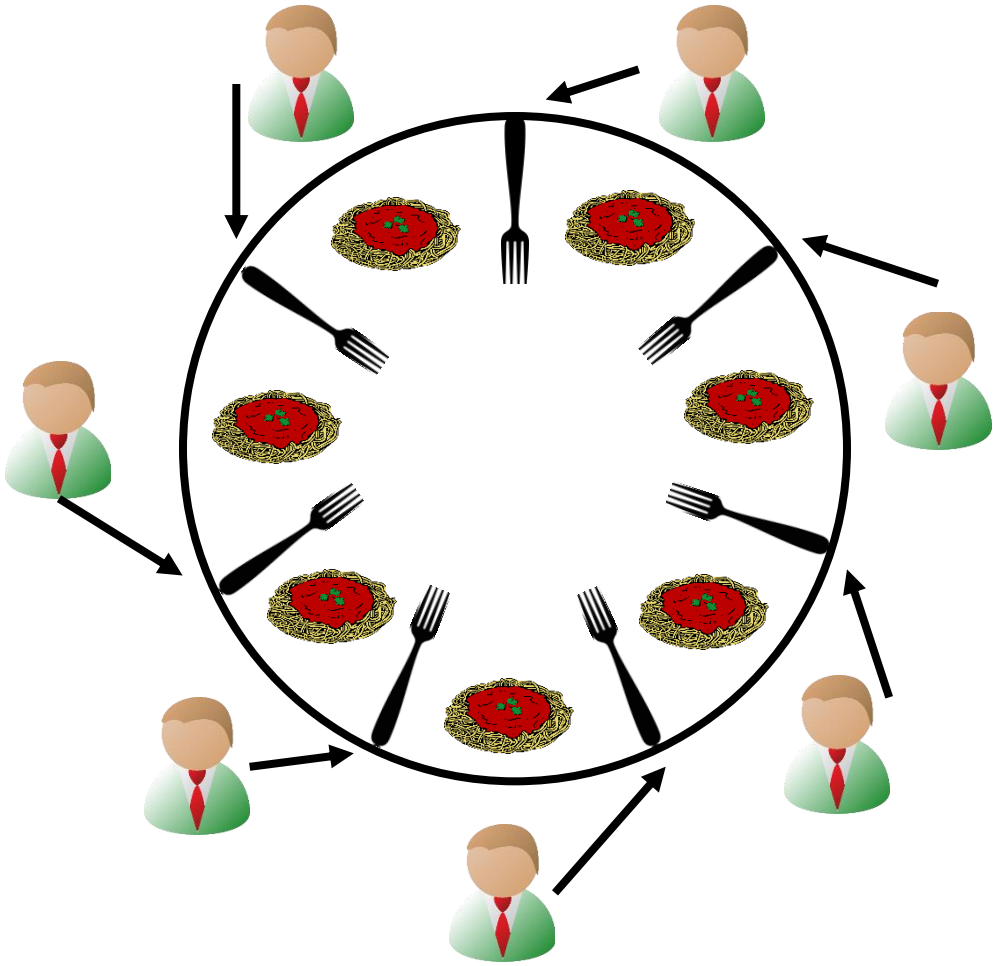
The forks are locks



Now he can eat. Then he will let go of the forks (locks). This means someone else can take them.

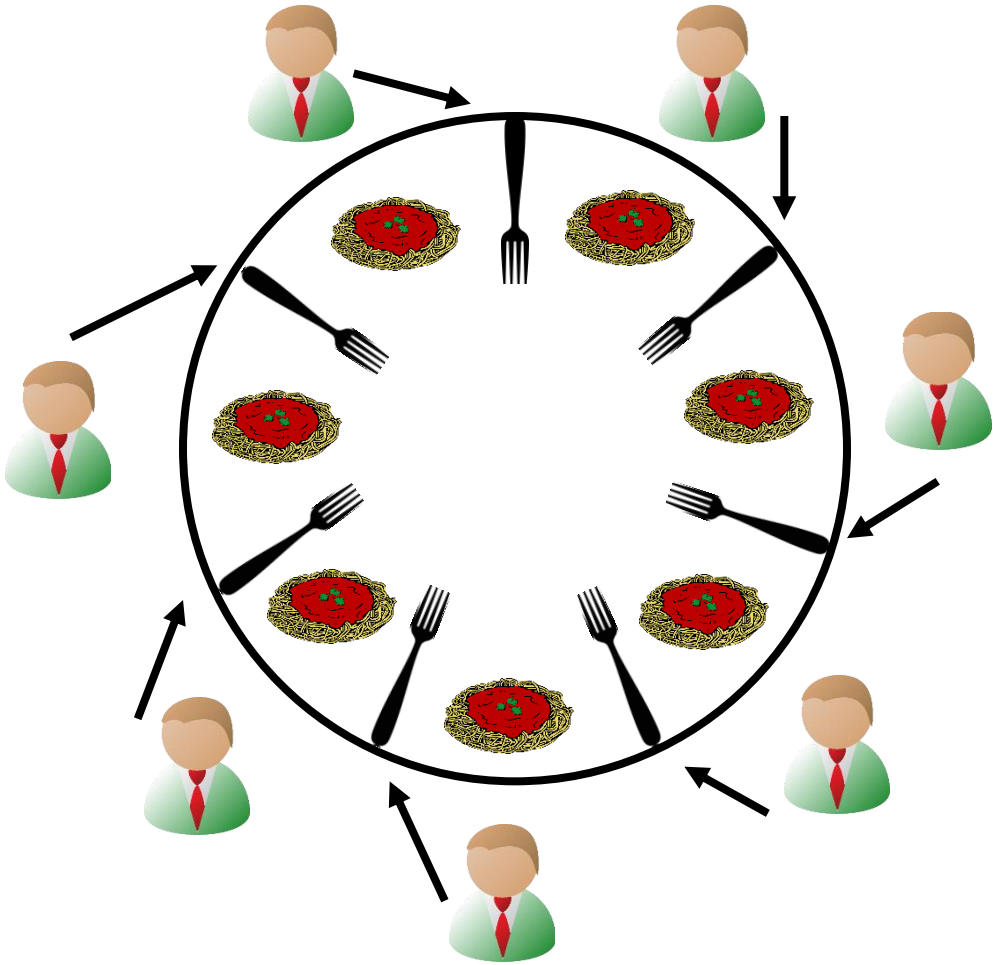
Dinning Philosophers Problem Dead-Lock

If they all get the right fork (lock) first, they can all get stuck (we have a dead-lock)



Dinning Philosophers Problem Dead-Lock

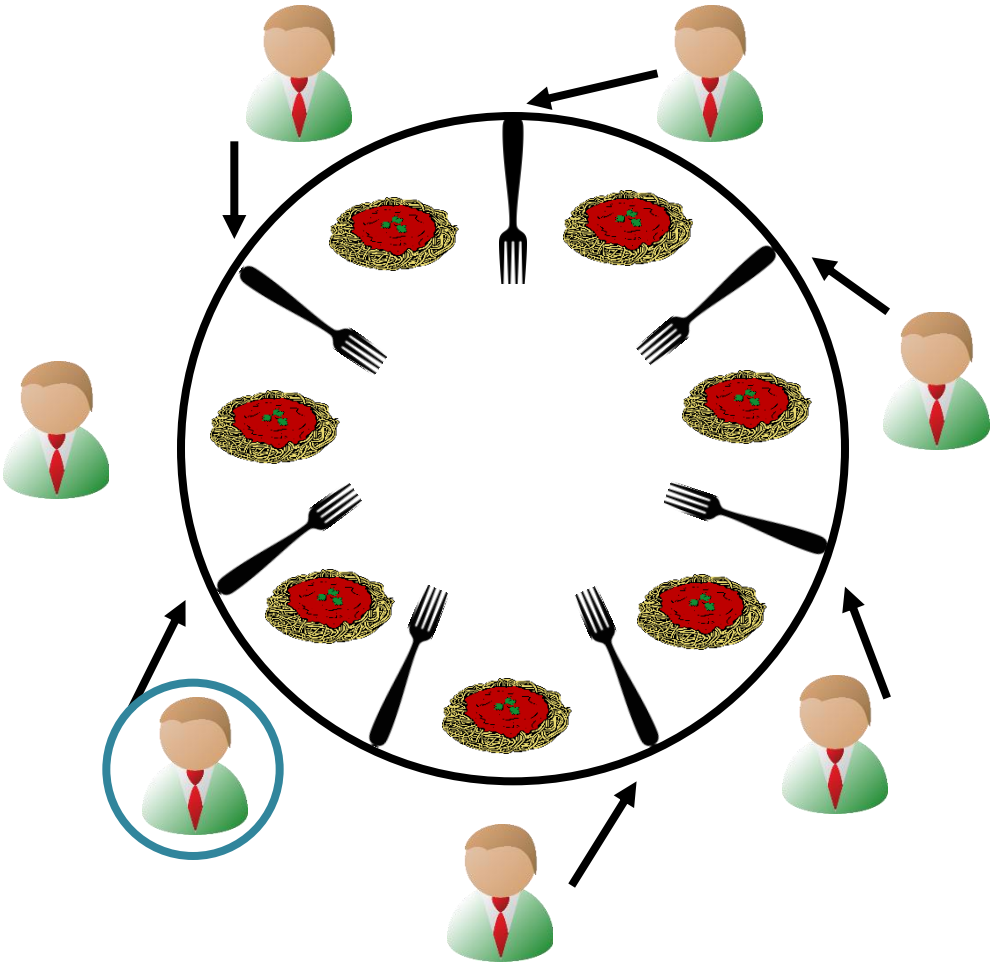
If they all get the left fork (lock) first, they can all get stuck (we have a dead-lock)



Dinning Philosophers Solution

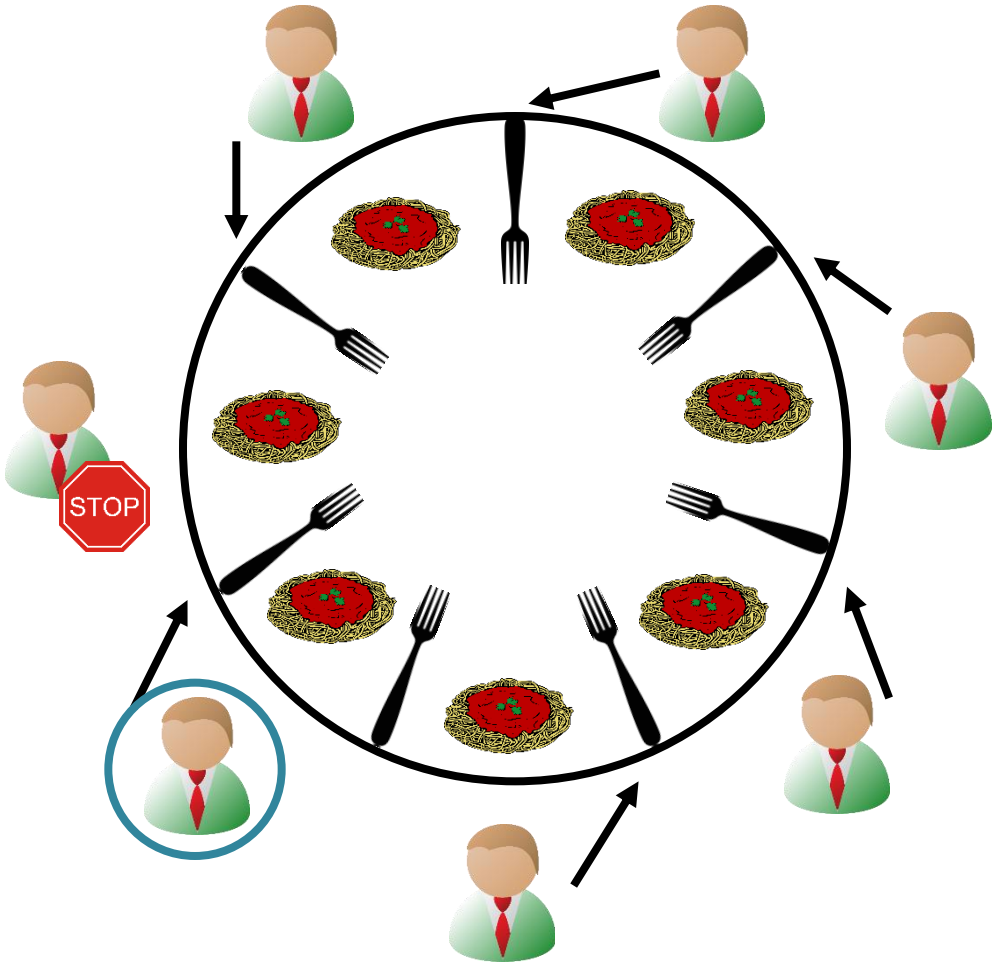
One of the philosophers takes the left fork (**lock**) **first** while the others take the right fork (**lock**) **first**.

Dinning Philosophers Solution

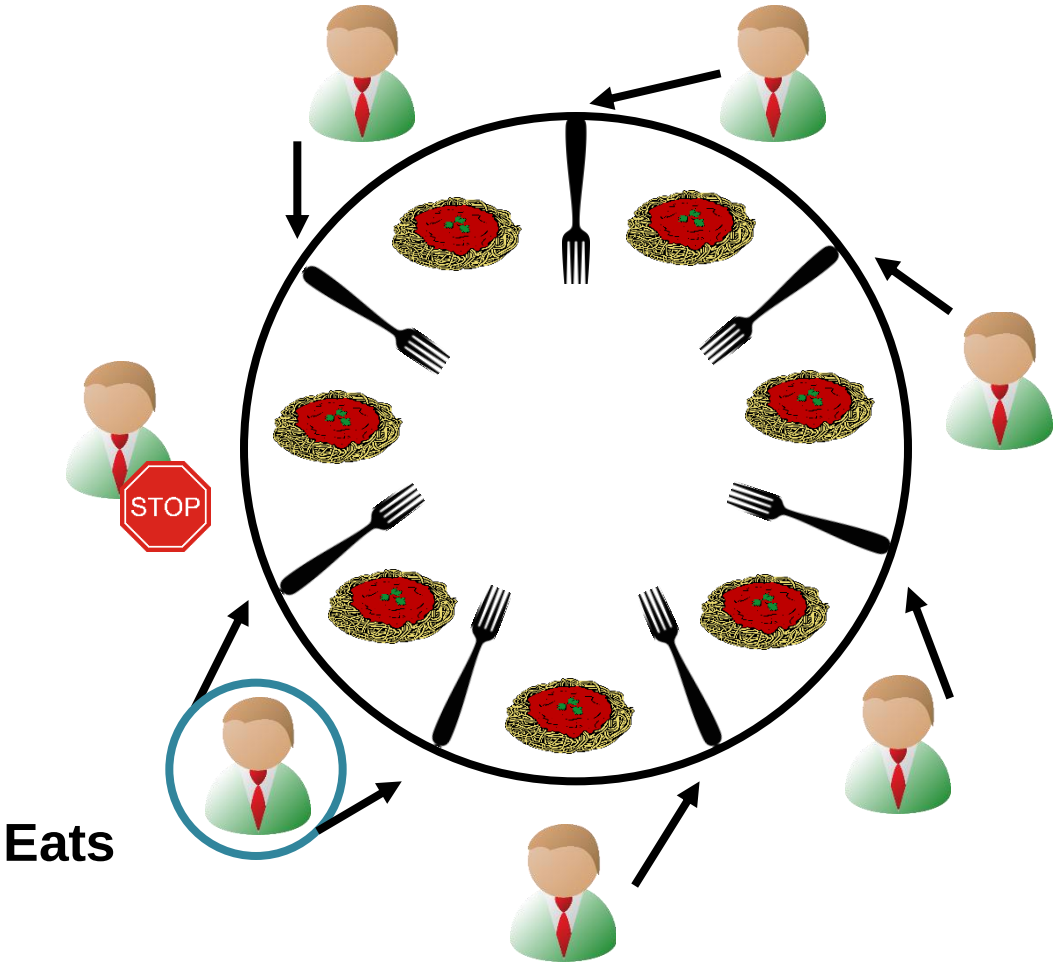


Dinning Philosophers Solution

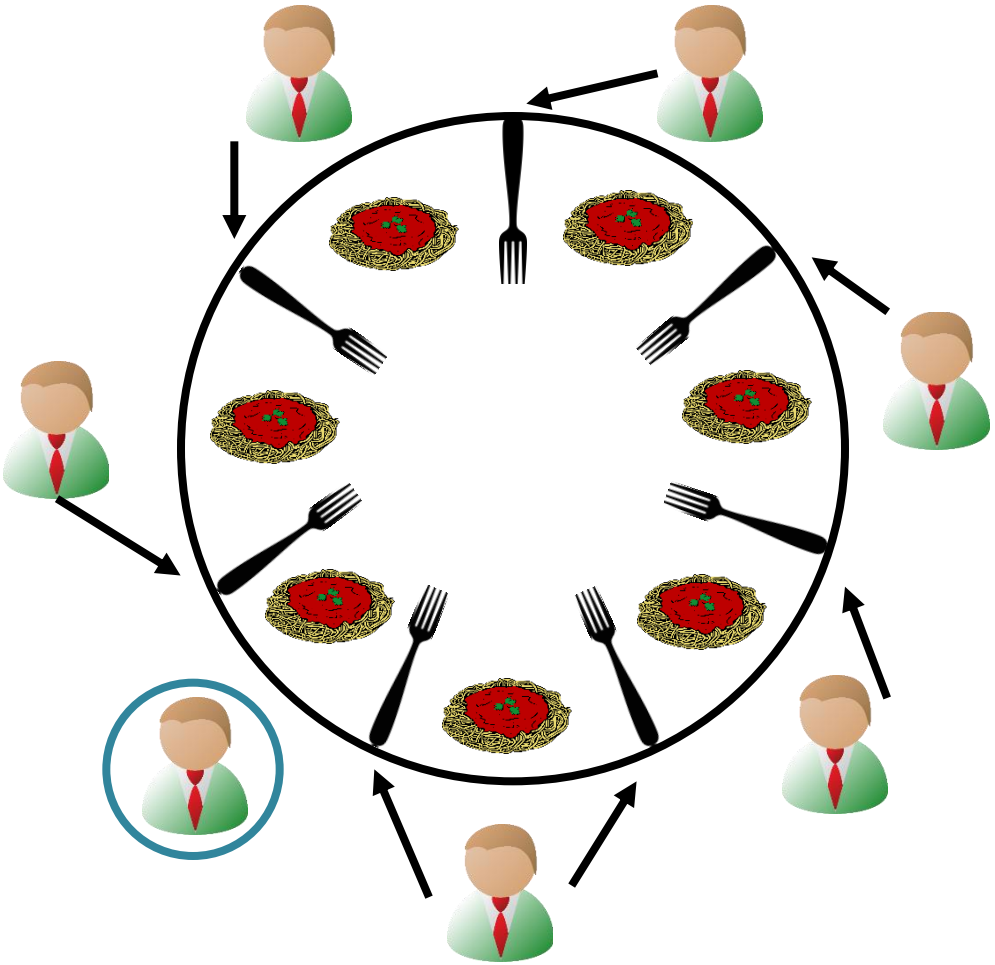
Can not take any fork (lock)



Dinning Philosophers Solution

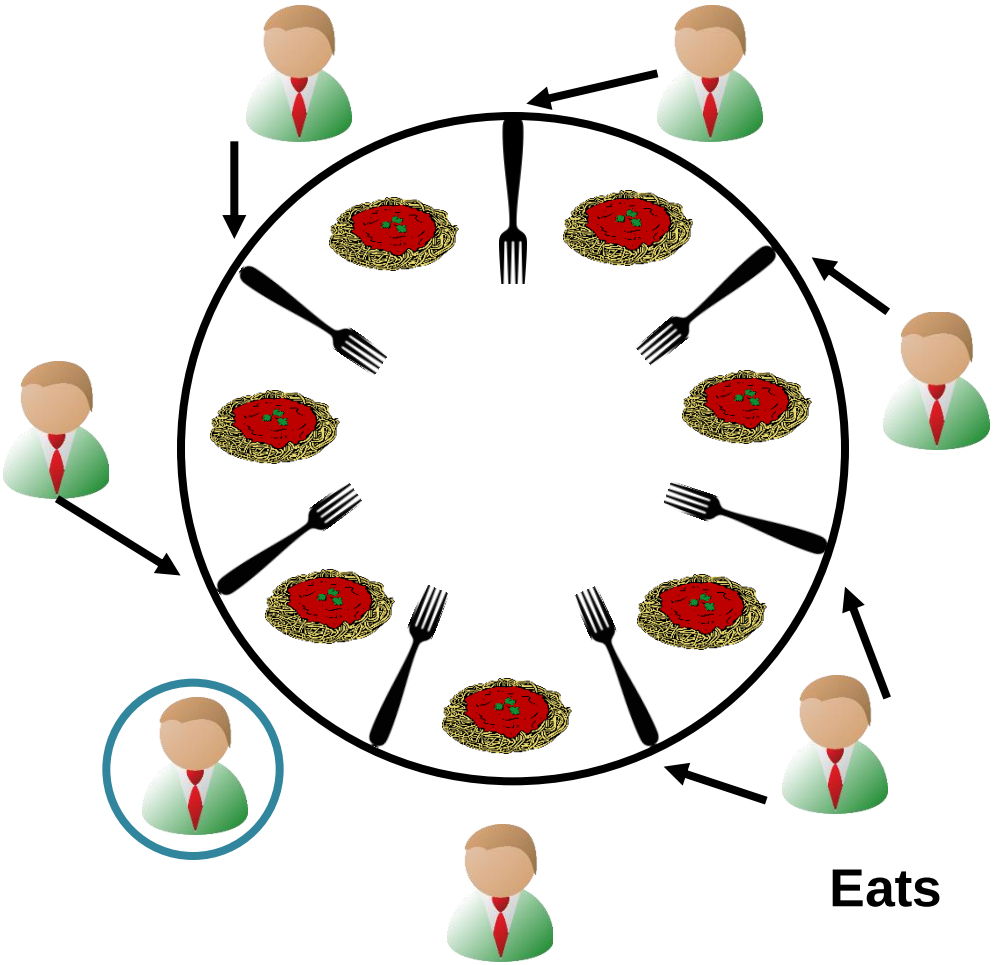


Dinning Philosophers Solution



Eats

Dinning Philosophers Solution



Dinning Philosophers Problem in real life

We have a server that runs a social network platform.
Each of our threads handles requests from a user.
It is possible for a group of users to be friends in a circle (like the philosophers sitting at the table).

The users can request sending a picture received from one of their friends to another.
In order to send the picture the thread needs to take the lock on the communication queue between A and B and between B and C.
If all people request a picture transfer at the same time the server will enter a dead-lock.