Deadlocks

- Acquire first
- Use lock
- Wait
- Release

Necessary Conditions
- Mutual Exclusion
- Hold and Wait
- No Preemption
- Circular Wait
- Process can be 2 phases - Resource Instance
- Resource can have multiple instances
- Resource
  - 2 kinds of modes - process
  - Resource Allocation Graph (RAG)
- xes chill, maybe decadock
  - no chill, no decadock
  - decadock → no, maybe

Chief in a bag?
If all resources have one instance

cycle is a sufficient condition for deadlock.
3 determination & reasoning

2 accordance

What to do about deceivers?
defrush
- circular motion
- hold a match
- no present
- must

I consider two not both
- ensure at least 1 of the necessary

Dysphagia prevention
Despite previous [no named reference]

In resource are shameful

Reduction in need
Can I mean say

As far as I know, for

all resources are portable

[No] no precondition

②
Sequences

(no nested unless held)

↓

e.g. do not do P(s) or P(S)

and reference as before getting used

goal idea in parenthesis

[no held & wait]
- unlock both

- book both

- find next freight lock

- find freight and NKC lock

Book plan to NKC via Cunard

- acquire reservations

- duplicate plan
ordered resource allocation

held a week or so. But no success.

- maybe or no progress at all.

[No] Cecelia went
if all the resources are available
and resources
0 -- -- -- -- (\(=-i\)) in resource
all resources are numbered
Suppose ordered resource allocation does not work for a deadlock.

Let $\nu_1$ (but why?)
P$_2$ case $r_2$
P$_2$ want $r_2$
P$_2$ case $r_2$

Exist Changed Case?
left out of scope proposed

A route, please
Each step counts.

Bouwers Algorithm

Superhappy version.
Customer

Bonuses

Repay

Interest

Time
Acquisition Plan
Credit Hunt $ predicted revenue

Revenue $ revenue

Become a subscriber &

Current $ share of voice
To each bowler, 70.

200

\[ \text{total} \]

\[ \text{count} = 50 \]

\[ \text{result} = 50 \]

\[ \text{each} \]

\[ \text{total} \]

\[ \text{result} \]

\[ \text{each} \]

\[ \text{total} \]

\[ \text{result} \]