



SOFTWARE • PRODUCTIVITY • GROUP



Falhas de Vivacidade

Fernando Castor

Professor Adjunto

Centro de Informática

Universidade Federal de Pernambuco

castor@cin.ufpe.br

Elaborado conjuntamente por:

Benito Fernandes, João Paulo Oliveira e Wesley Torres



UNIVERSIDADE FEDERAL
DE PERNAMBUCO

cin.ufpe.br

Liveness (vivacidade)

- A execução de um programa em algum momento chega a um **estado desejável**
- **Algo de bom** acontecerá durante a execução do programa
 - Um processo que pode executar será executado
 - Uma mensagem enviada de um processo a outro será recebida

Falhas de Liveness

- Difíceis de identificar e evitar durante o projeto
- Possíveis Falhas
 - Deadlock
 - Starvation
 - Livelock

Deadlock

- Duas ou mais threads são bloqueadas para sempre, esperando uma pela outra.
 - Ou as threads estão bloqueadas esperando por um evento que nunca irá acontecer.

Deadlock

- Duas ou mais threads são bloqueadas para sempre, esperando uma pela outra.
 - Ou as threads estão bloqueadas esperando por um evento que nunca irá acontecer.
- Exemplo
 - Duas pessoas desenhando diagramas
 - usando apenas um lápis e uma régua
 - Se cada uma pegar um item e ambas forem teimosas, ocorre um **deadlock**

Condições **necessárias** (mas não **suficientes**) para a ocorrência de deadlocks

1. **Exclusão mútua:** apenas um processo de posse de cada recurso
2. **Segura e espera:** um processo pode adquirir outros recursos enquanto segura um
3. **Sem preempção:** não é possível remover um recurso de um processo à força
4. **Espera circular:** um processo depende dos recursos de outro processo e vice-versa

Starvation

- Situação na qual uma thread **não tem acesso a um recurso compartilhado**
 - não pode prosseguir sua execução
- Recurso compartilhado não disponível por um longo período

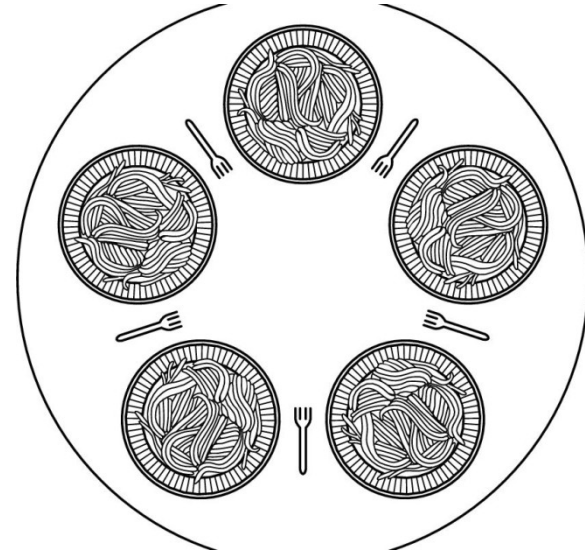


Problema: Jantar dos Filósofos

- Problema clássico da computação concorrente
- Representa a competição por um conjunto de recursos, nesse caso os garfos
- Os filósofos não podem comer todos ao mesmo tempo por não possuir garfos o suficiente
- Necessário garantir que os filósofos não entrem em deadlock, **livelock** ou starvation

Jantar dos Filósofos - enunciado

- Sentados em círculos
- Filósofos comem / pensam
- Para comer é necessário 2 garfos
- Escolher um garfo por vez



Quais propriedades
uma solução para
esse problema
deve satisfazer?

Implemente uma
simulação de jantar dos
filósofos. A cada 10ms,
cada filósofo joga uma
moeda para saber se
vai comer ou não

Uma solução simples:

- Filósofos são muito civilizados
 - Devolvem o talher se não puderem pegar os dois
- Quebra a Condição Necessária 2
 - Segura e Espera
- **Problema:** Livelock



Outra solução simples:

- Filósofos são muito organizados
 - Talheres são numerados de 1 a 5
 - Um filósofo sempre pega primeiro o talher mais próximo de menor número
- Quebra a Condição Necessária 4
 - **Espera Circular**