



Universidade Tecnológica Federal do Paraná
Departamento Acadêmico de Computação
Bacharelado em Ciência da Computação

Sistemas Operacionais

Impasse e Inanição

Prof. Rodrigo Campiolo

23/10/19

Introdução

- Recursos
 - Exemplos de recursos: área de memória, impressora, estrutura de dados.
 - Processos precisam de acesso a recursos em uma determinada ordem.
 - Exemplo de problema:
 - Processo P está usando um recurso A.
 - Processo Q está usando um recurso B.
 - Processo Q solicita o recurso A e é bloqueado.
 - Processo P solicita o recurso B e é bloqueado.

Introdução

- Definições
 - Impasse (**Deadlock**)
 - Dois ou mais processos aguardam eventos que dependem dos outros processos.
 - Inanição ou adiamento indefinido (**Starvation**)
 - Um processo é negligenciado continuamente pelo escalonador, assim não consegue proceder sua execução.
 - **Livelock**
 - Dois ou mais processos alteram seus estados continuamente sem fazer trabalho útil em função de alterações de outros processos .

Impasses

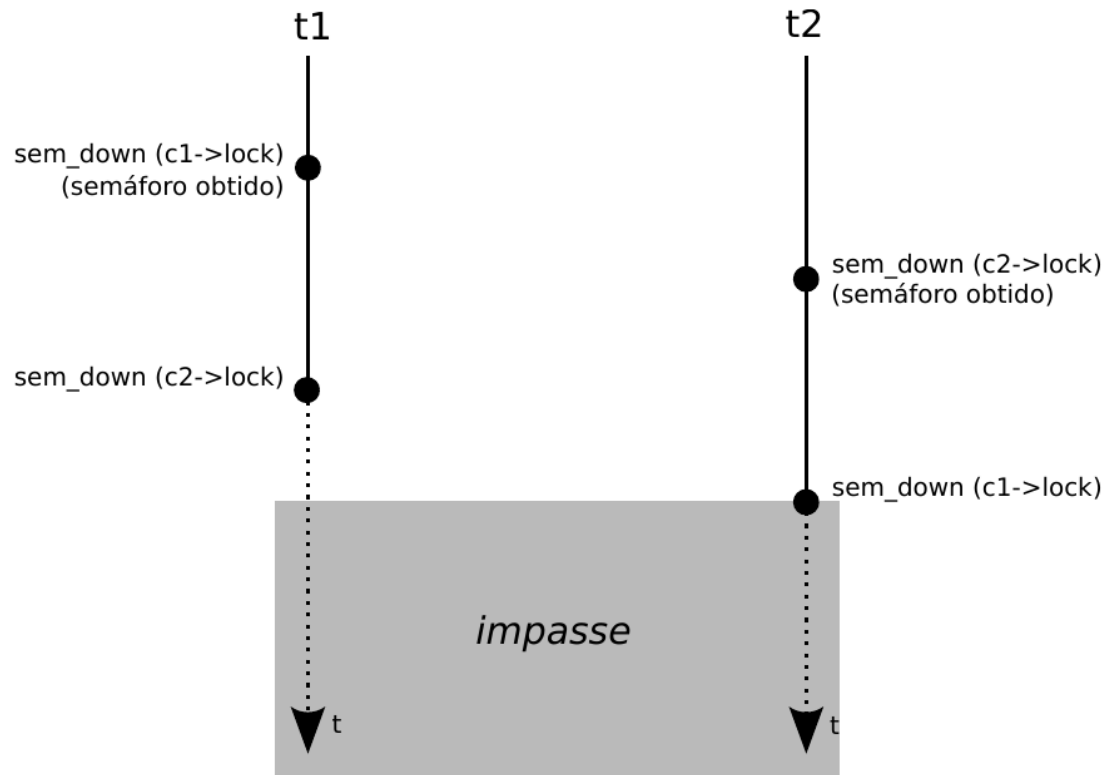
- Definição formal (Tanenbaum):

Um conjunto de processos estará em situação de impasse se cada processo no conjunto estiver esperando por um evento que apenas outro processo no conjunto pode causar.

Impasses

- Exemplo:

Impasse na transferência entre contas



Fonte: Maziero

Impasses

- Quatro condições obrigatórias:
 - Exclusão mútua
 - Todo recurso está associado exclusivamente a um processo.
 - Posse e espera
 - Processos podem manter recursos e solicitar novos recursos.
 - Não preempção
 - Recursos não podem ser preemptados, isto é, tomados de um processo.
 - Espera circular
 - Processos formam uma cadeia circular de posse e espera por um recurso.

Impasses

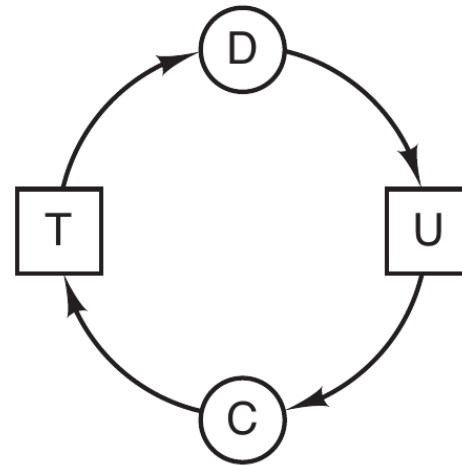
- Grafo de Alocação de Recursos



(a)



(b)



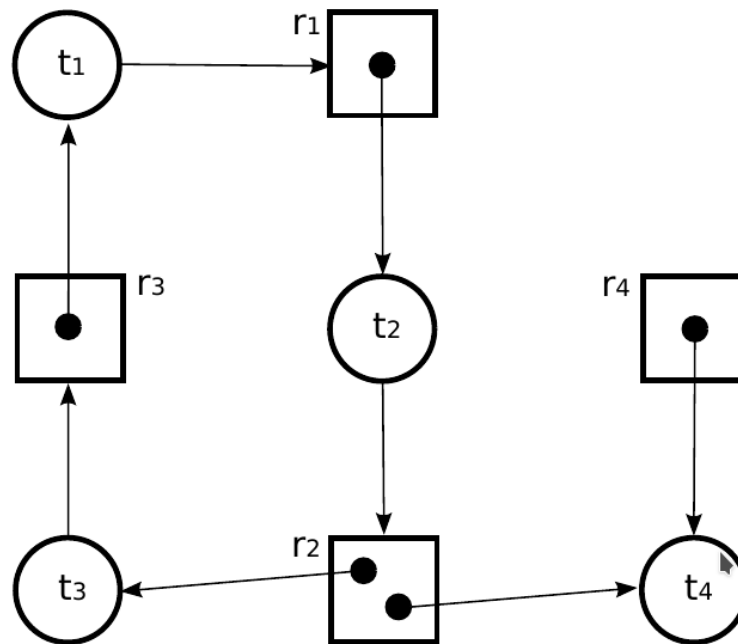
(c)

Grafos de alocação de recursos:
(a) processo de posse de um recurso.
(b) Solicitação de um recurso.
(c) Impasse.

Fonte: Tanenbaum

Impasses - Questão?

- Observando o grafo, há um impasse?



Fonte: Maziero

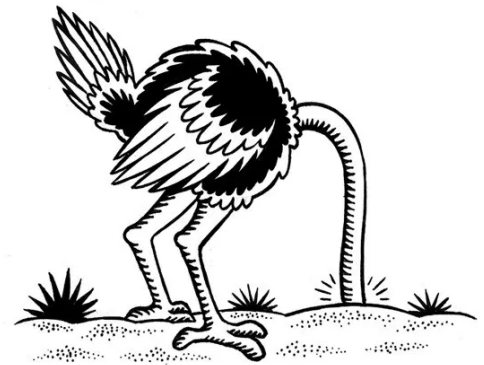
t: tarefa
r: recurso

Impasses

- Técnicas de tratamento de impasses
 - Algoritmo do Avestruz
 - Prevenção de impasses
 - Negar ao menos uma das condições de impasse.
 - Evitação de impasses
 - Alocar cuidadosamente os recursos para evitar impasses.
 - Detecção e recuperação de impasses
 - Detectar o impasse e aplicar medidas para resolvê-lo.

Impasses - Técnicas de Tratamento

- Algoritmo do Avestruz
 - Ignora a existência do problema de impasses.
 - Unix e Windows adotam essa técnica para lidar com os impasses.
 - Poucos impasses x custo de prevenção.



Fonte: pitangua.com.br

Impasses - Técnicas de Tratamento

- Prevenção de impasses
 - Negar a **exclusão mútua**.
 - Exemplo: técnica de *spooling*.
 - Negar a **posse e espera**.
 - Usar apenas um recurso por vez.
 - Obter todos os recursos antes de executar.
 - Estabelecer um limite de tempo para posse e espera.

Impasses - Técnicas de Tratamento

- Prevenção de impasses
 - Negar a **não-preempção**.
 - Retirar um recurso de um processo.
 - Interessante para situações que o estado do processo pode ser salvo e restaurado.
 - Negar a **espera circular**.
 - Ordenação de todos os recursos do sistema.
 - Processos solicitam recursos somente nessa ordem.

Impasses - Técnicas de Tratamento

- Evitação de impasses
 - Consiste em evitar o impasse pela alocação cuidadosa de recursos.
 - Estado seguro x estado inseguro.
 - Um estado é seguro se a partir dele é possível realizar a alocação de recursos aos processos e evitar assim o impasse.

Estado Seguro

| | posse | total |
|----------------|-------|-------|
| P _A | 3 | 9 |
| P _B | 2 | 4 |
| P _C | 2 | 7 |

Disponível: 3 recursos

Estado Inseguro

| | posse | total |
|----------------|-------|-------|
| P _A | 3 | 9 |
| P _B | 2 | 4 |
| P _C | 2 | 7 |

Disponível: 2 recursos

Impasses - Técnicas de Tratamento

- Evitação de impasses
 - Algoritmo do Banqueiro
 - Criado por Dijkstra (1965)
processos = clientes de banco
recursos = créditos
banqueiro = SO
 - Decidir a alocação de créditos para manter em um estado seguro.
 - Na prática é de difícil implementação, pois necessita conhecer previamente os recursos e ordem das solicitações.

Impasses - Técnicas de Tratamento

■ Algoritmo do Banqueiro

| | Unidades de fita | Plotters | Scanners | Blu-rays |
|-----------------------------------|------------------|----------|----------|----------|
| $E = (4 \quad 2 \quad 3 \quad 1)$ | | | | |
| $A = (2 \quad 1 \quad 0 \quad 0)$ | | | | |

Situações:

- **P3** solicita 1 plotter.
- **P1** solicita 1 fita.

Matriz alocação atual

$$C = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 1 & 2 & 0 \end{bmatrix}$$

Matriz de requisições

$$R = \begin{bmatrix} 2 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix}$$

E: recursos existentes

A: recursos disponíveis

Matrizes: cada linha representa os recursos para um processo.

Impasses - Técnicas de Tratamento

- Detecção e Recuperação
 - **Detecção** por inspeção ao grafo de alocação de recursos.
 - O grafo deve ser mantido pelo sistema, isto é, atualizado nas alocações e liberações de recursos.
 - Executar periodicamente um algoritmo para detecção de ciclos no grafo.
 - **Problema:** custo computacional do algoritmo.

Impasses - Técnicas de Tratamento

- Detecção e Recuperação
 - **Recuperação**
 - Eliminar tarefas:
 - Eliminar todas ou selecionar tarefas.
 - Critérios para eliminar tarefas.
 - Reversão de estado:
 - Gravar o estado periodicamente dos processos e retomar a partir de um estado seguro.

Impasses - Conclusões

- Custo x Correção
- Impasses na comunicação
- Pesquisas na área

Referências

- OLIVEIRA, R. S. et al. **Sistemas operacionais**. 4ª Edição e Slides online. Bookman. 2010.
- SILBERSCHATZ, A.; GALVIN, P. B.; GAGNE, G. **Fundamentos de sistemas operacionais**. 9. ed. LTC, 2015.
- TANENBAUM, Andrew S. BOSS Helbert. **Sistemas operacionais modernos**. 4. ed. São Paulo, Pearson Education do Brasil, 2016.
- MAZIERO, Carlos A. **Sistemas operacionais: conceitos e mecanismos**. Online. 2019.

Referências

- OLIVEIRA, R. S. et al. **Sistemas operacionais**. 4ª Edição e Slides online. Bookman. 2010.
- SILBERSCHATZ, A.; GALVIN, P. B.; GAGNE, G. **Fundamentos de sistemas operacionais**. 9. ed. LTC, 2015.
- TANENBAUM, Andrew S. BOSS Helbert. **Sistemas operacionais modernos**. 4. ed. São Paulo, Pearson Education do Brasil, 2016.
- MAZIERO, Carlos A. **Sistemas operacionais: conceitos e mecanismos**. Online. 2019.