

Processamento de Transações

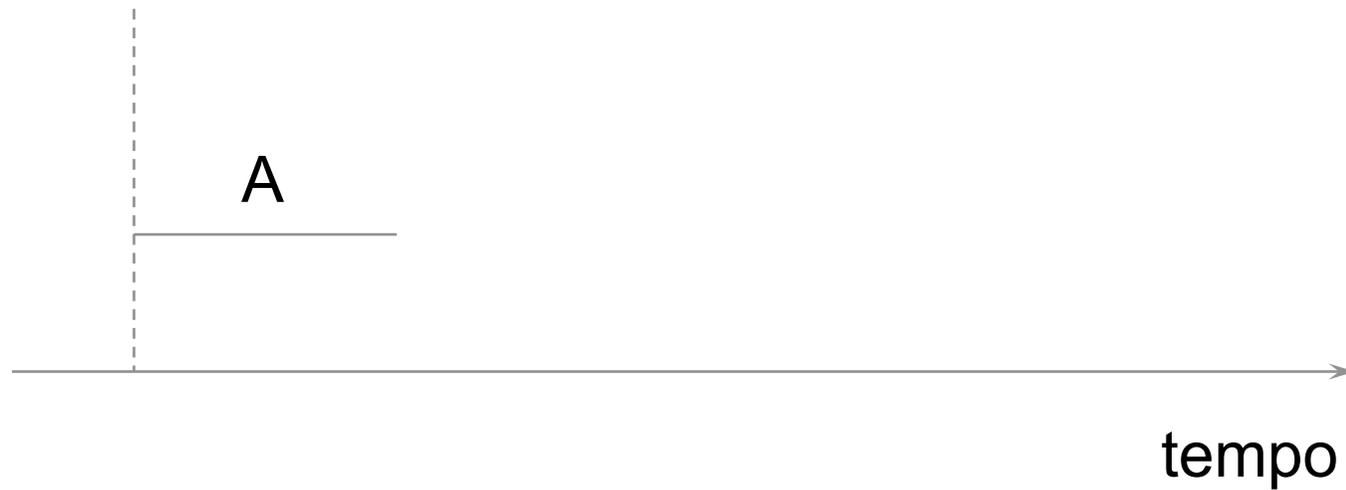
Laboratório de Bases de Dados

Profa. Dra. Cristina Dutra de Aguiar Ciferri

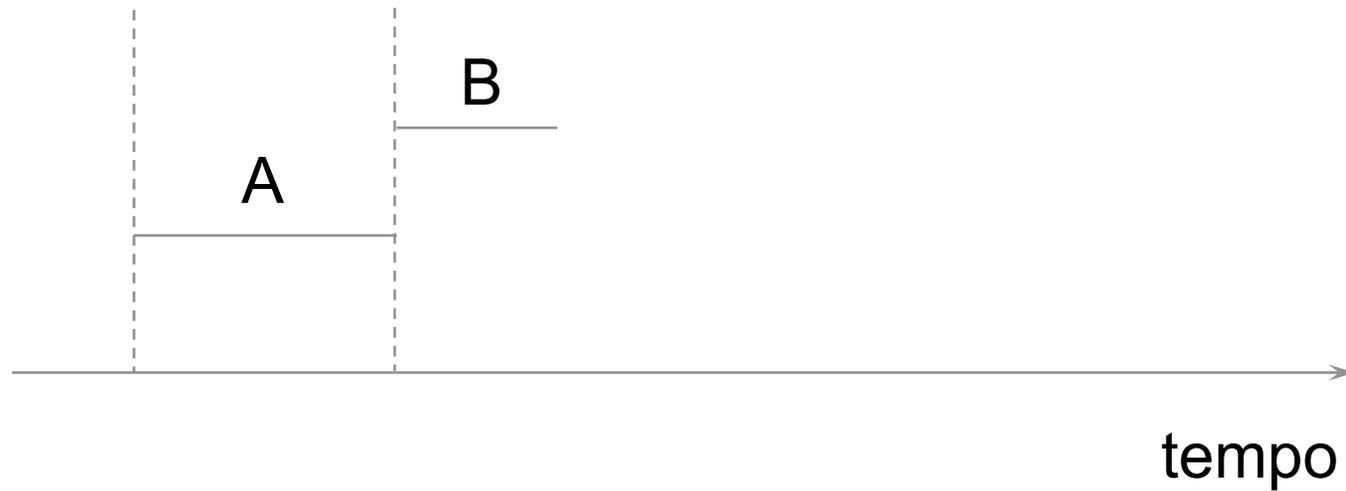
Introdução

- Ambiente multiusuário
 - vários usuários utilizam o mesmo sistema ao mesmo tempo
 - múltiplos programas (transações) compartilham a mesma CPU
- Forma de execução dos programas
 - intercalada (*interleaved*)
 - *alguns comandos de um programa são executados e o programa é suspenso; alguns comandos de outro programa são executados, o programa é suspenso, e assim por diante ...*

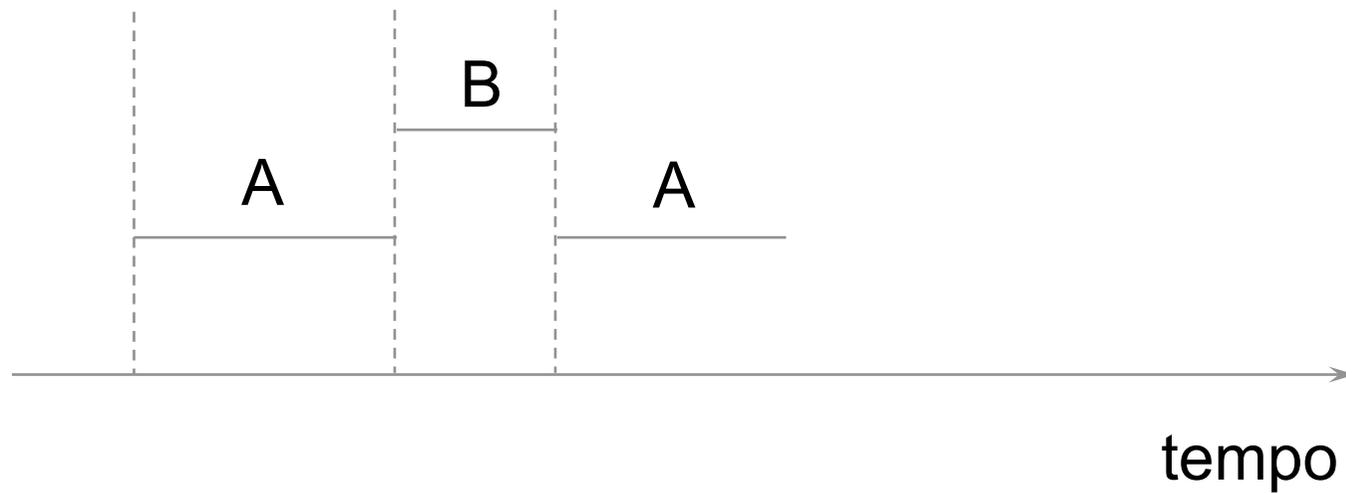
Execução Intercalada



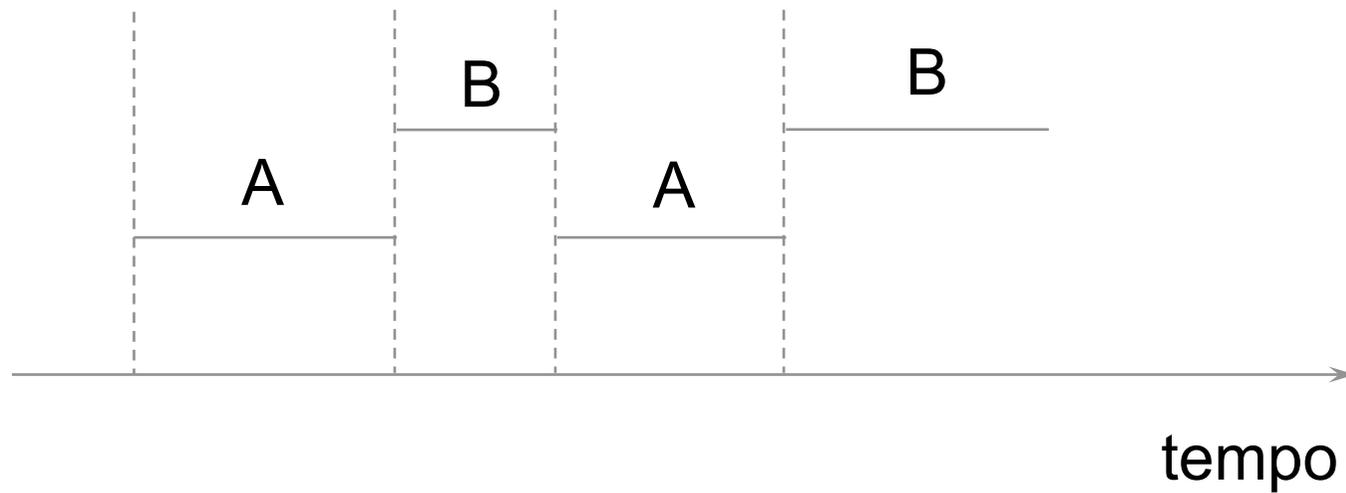
Execução Intercalada



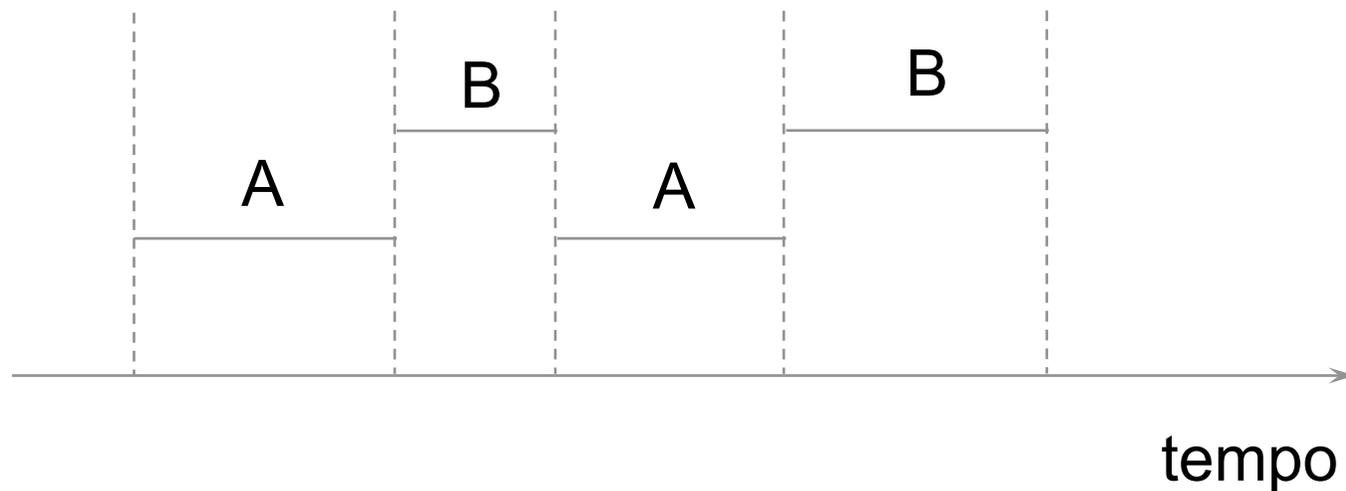
Execução Intercalada



Execução Intercalada



Execução Intercalada



- SBD multiusuário
 - recursos principais: dados armazenados no BD

Transação

- Unidade de execução de um programa que acessa ou altera o conteúdo do BD
- Seqüência de operações de escrita/leitura no BD
- Características
 - é delimitada por declarações da forma **início da transação** e **fim da transação**
 - todas as operações de escrita/leitura entre essas duas declarações são consideradas parte de uma mesma transação

Transação

- Características
 - um programa de aplicação pode conter mais do que uma transação, se as operações de escrita/leitura são limitadas por diferentes pares de declaração **início da transação** e **fim da transação**
- Observações
 - os dados do BD estão armazenados em memória secundária
 - as operações que não sejam leitura/escrita não apresentam efeito para o BD

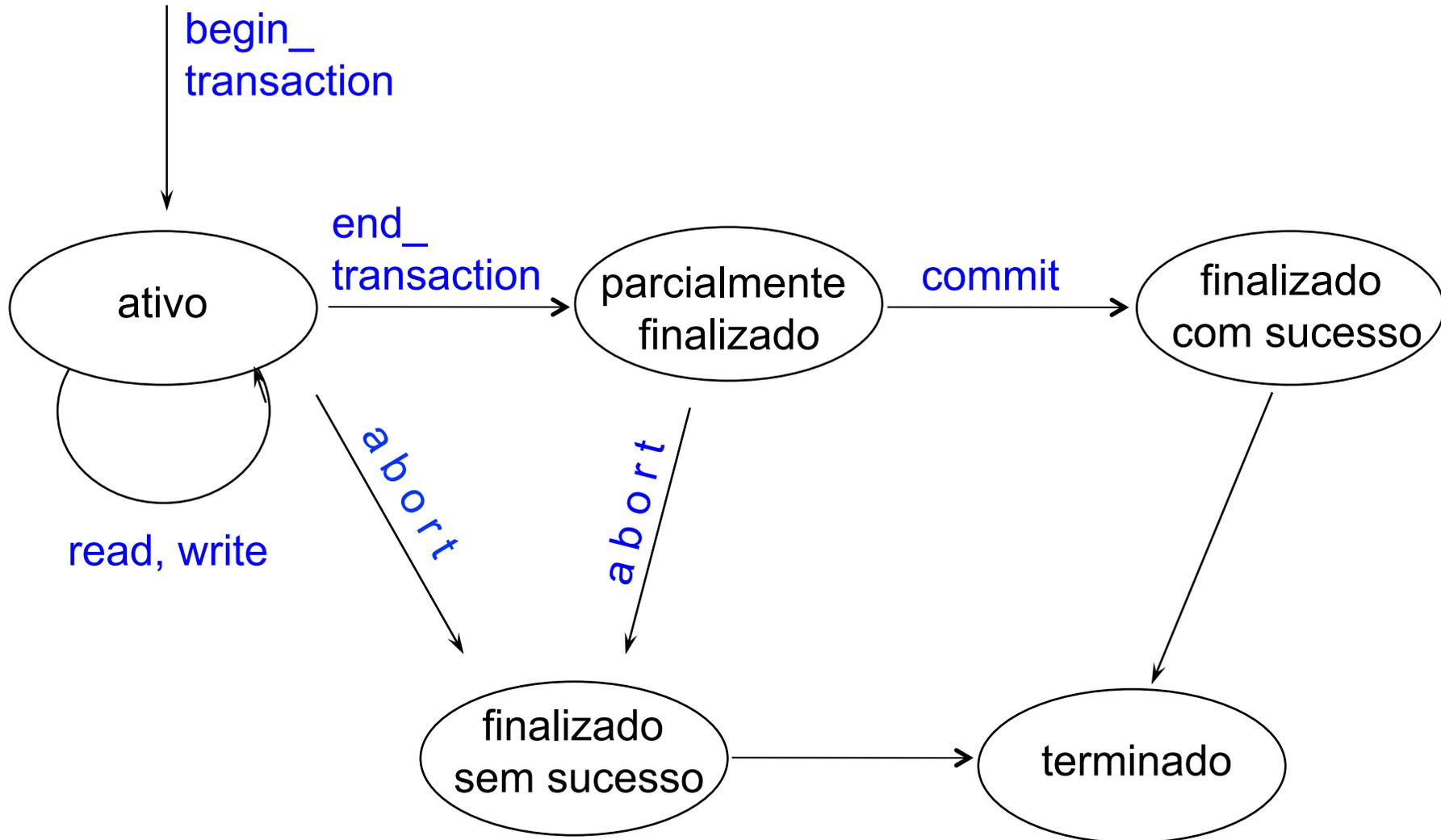
Operações das Transações

- `begin_transaction`
 - início da execução de uma transação
- `read` ou `write`
 - operações de leitura ou escrita nos dados do BD
- `end_transaction`
 - final da execução de uma transação
 - deve ser verificado se a transação executará `commit` ou `abort`

Operações das Transações

- `commit_transaction`
 - indica que a transação foi finalizada com sucesso
 - torna permanente as alterações realizadas no BD
- `abort_transaction (rollback)`
 - indica que a transação foi finalizada sem sucesso
 - descarta as alterações já realizadas no BD

Transição de Estado



Operação de Leitura

- Representada por $r(x)$ ou `read_item(x)`
- Lê o item de dado x na variável de programa x
- Passos
 - encontrar o endereço do bloco que contém x
 - copiar o bloco para o *buffer* da memória principal (se necessário)
 - copiar o valor de x do *buffer* para a variável x

Operação de Escrita

- Representada por $w(x)$ ou `write_item(x)`
- Escreve o valor da variável de programa x no item de dado x
- Passos
 - encontrar o endereço do bloco que contém x
 - copiar o bloco para o *buffer* da memória principal (se necessário)
 - copiar o valor da variável x para o *buffer*
 - escrever o novo valor do item de dado x no disco (atualização do BD)

Observações

- Transações submetidas pelos usuários
 - podem executar concorrentemente
 - podem acessar e alterar os mesmos itens de dados
- Execução não controlada
 - pode originar problemas
 - **inconsistência** do banco de dados

Exemplos

- Transação 1

$r(x)$

$x := x - n$

$w(x)$

$r(y)$

$y := y + n$

$w(y)$

- Transação 2

$r(x)$

$x := x + m$

$w(x)$

- $t_2: r_2(x) w_2(x) c_2$

- $t_1: r_1(x) w_1(x) r_1(y) w_1(y) c_1$