

Sistemas Operacionais

Aula 08 – Comunicação entre Tarefas

Prof. Igor da Penha Natal

Conteúdo

- 1 Cooperação entre tarefas
- 2 Comunicação direta ou indireta
- 3 Sincronismo
- 4 Formato de envio
- 5 Capacidade dos canais
- 6 Confiabilidade dos canais
- 7 Número de participantes

Cooperação entre tarefas

Cooperar: trabalhar em conjunto para atingir um objetivo.

Por que construir sistemas com tarefas cooperantes?

- **Atender usuários simultâneos:** servidor de banco de dados ou de e-mail.
- **Computadores multicore:** acelerar a execução.
- **Modularidade:** dividir sistema em módulos autônomos.
- **Aplicações interativas:** navegadores Web e editores usam *threads* para melhorar interatividade.

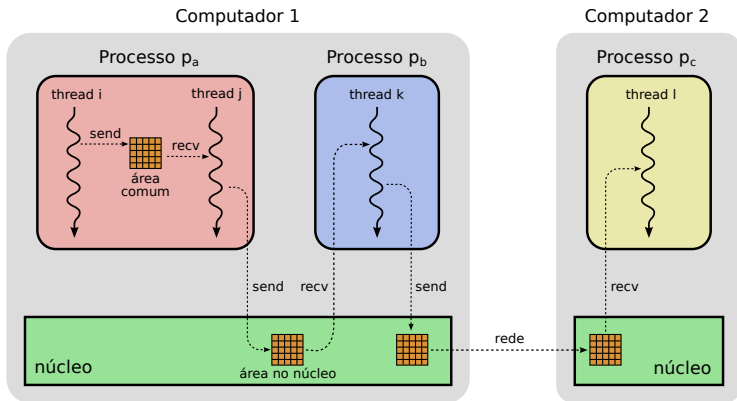
Cooperação entre tarefas

$$\text{Cooperação} = \left\{ \begin{array}{l} \text{Comunicação} \\ + \\ \text{Coordenação} \end{array} \right.$$

Comunicação: troca de informações entre tarefas

Coordenação: ordem e dependência das ações das tarefas

Escopo da comunicação



Genericamente se usa “IPC - *Inter-Process Communication*”

Aspectos da comunicação

A comunicação pode ser:

- Direta ou indireta
- Síncrona ou assíncrona
- Por mensagens ou por fluxos
- Capacidade dos canais
- Confiável ou não-confiável
- Número de participantes

Comunicação direta ou indireta

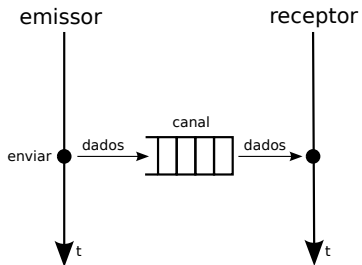
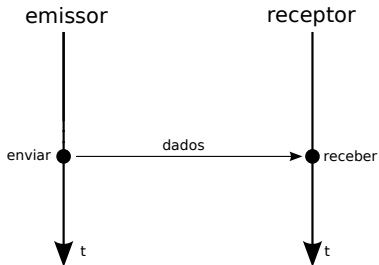
Comunicação **direta**:

- Emissor e receptor são identificados
- Enviar dados a uma tarefa: *enviar (destino, dados)*
- Receber dados de uma tarefa: *receber (origem, dados)*

Comunicação **indireta**:

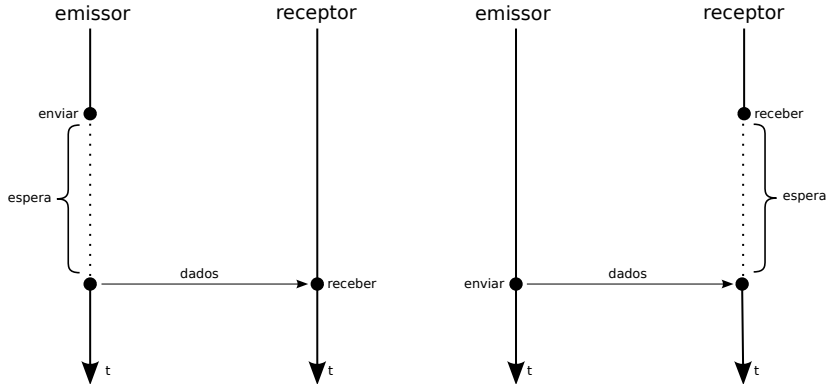
- Emissor e receptor comunicam através de um **canal**
- Enviar dados ao canal: *enviar (canal, dados)*
- Receber dados do canal: *receber (canal, dados)*

Comunicação direta ou indireta



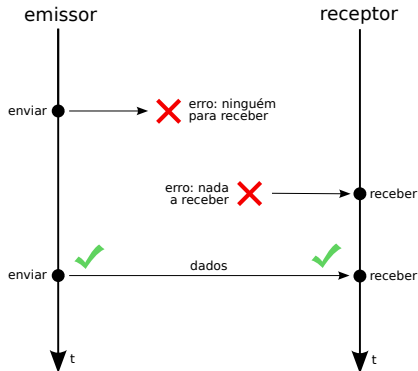
Comunicação síncrona (bloqueante)

As operações de envio/recepção **podem bloquear** as tarefas



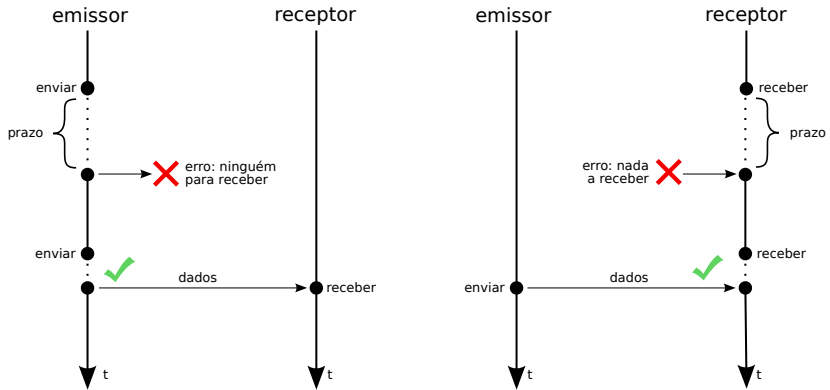
Comunicação assíncrona (não-bloqueante)

As operações de envio/recepção **não bloqueiam** as tarefas



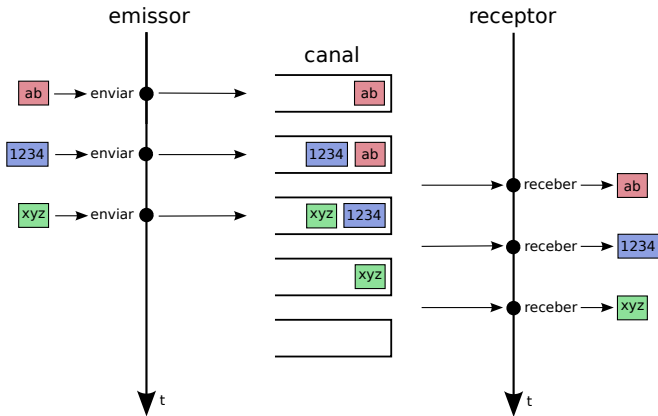
Comunicação semi-síncrona (semi-bloqueante)

As operações são bloqueantes durante um prazo predefinido



Formato da informação: mensagens

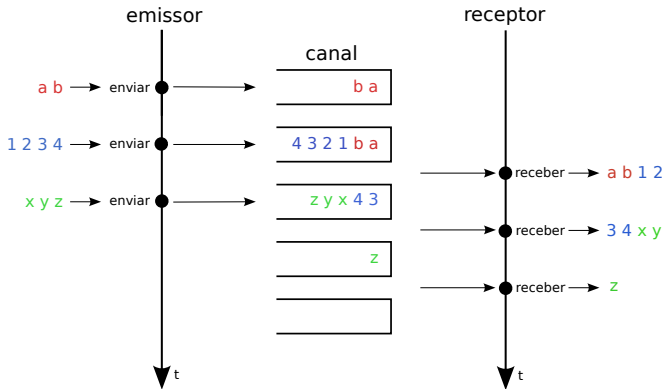
Mensagem: pacote de dados recebido pelo destino na íntegra.



Formato da informação: fluxo de dados

O canal de comunicação é visto como um arquivo.

O emissor “escreve” dados no canal e o receptor os “lê” do canal.



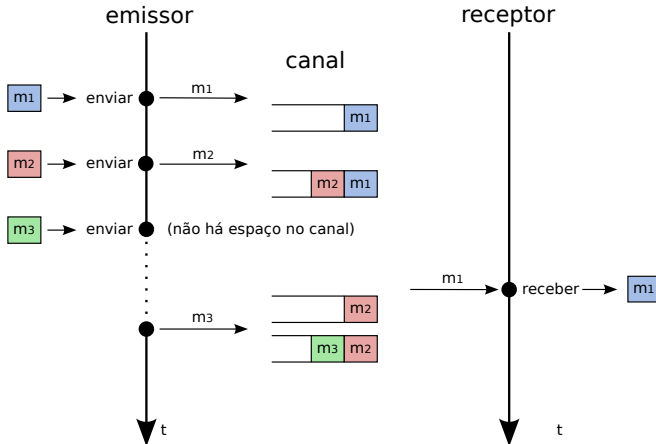
Capacidade dos canais

Capacidade do canal armazenar **dados em trânsito**:

- **Capacidade nula ($n = 0$)**: a comunicação é feita por transferência direta entre emissor e receptor.
- **Capacidade infinita ($n = \infty$)**: o emissor sempre pode enviar dados, que serão armazenados no *buffer* do canal enquanto o receptor não os consumir.
- **Capacidade finita ($0 < n < \infty$)**: o canal pode armazenar uma quantidade finita de dados.

Canal de capacidade finita

Comunicação **bloqueante** usando um canal com capacidade 2:

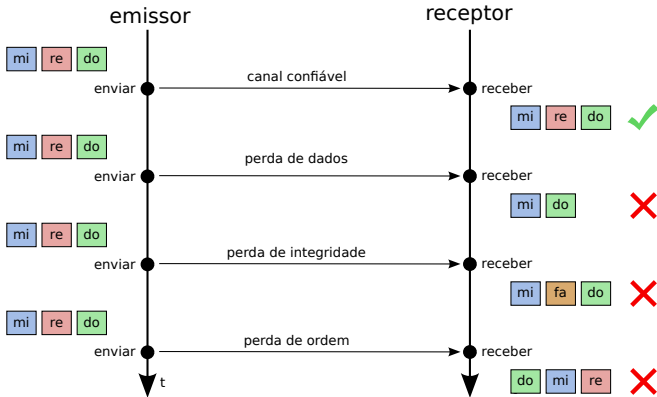


Confiabilidade dos canais

Um canal de comunicação pode ser:

- **Confiável**: transporta ao destino **todos os dados**, mantendo a **integridade** e **ordem** de envio.
- **Não-Confiável**: podem ocorrer perdas:
 - **De dados**: dados enviados podem não ser recebidos
 - **De integridade**: dados podem chegar alterados
 - **De ordem**: os dados podem chegar fora de ordem

Confiabilidade dos canais



Número de participantes

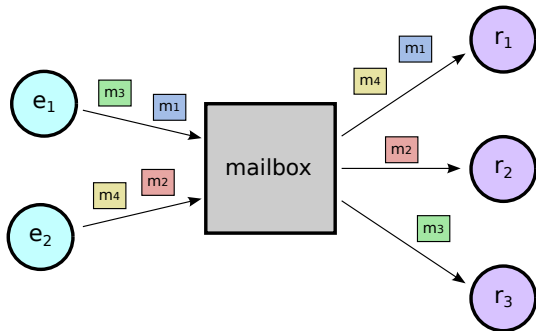
1:1 : um emissor e um receptor interagem através do canal de comunicação.

M:N : um ou mais emissores enviam mensagens para um ou mais receptores.

- Cada mensagem é recebida por **apenas um receptor** (*mailbox*)
- Cada mensagem é recebida por **todos os receptores** (*canal de eventos*).

Mailbox

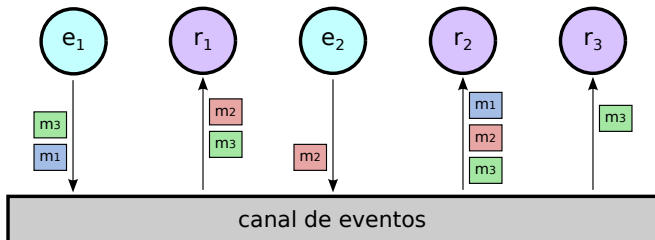
Cada mensagem tem um só receptor.



Modelo conhecido como *message queue*.

Canal de eventos

Cada mensagem pode ter vários receptores.



Modelo também conhecido com *publish/subscribe*.