

* Tipos de SO:

↳ Monoprogramados/Monotarefa: Só permite um programa ativo em um dado período de tempo, permanecendo na RAM até seu fim.
Exemplo: MS-DOS.

+ usados!

⚠️ ↳ Multiprogramados/Multitarefa: Permitem mais de um programa na memória, compartilhando tempo de CPU, dentre outros recursos.
Exemplo: UNIX, Windows, etc.

* SOs monoprogramados

- Caracterizado pela dedicação total dos recursos computacionais a um único programa. → frequentemente ociosos!
- Recursos mal utilizados, mas implementados com facilidade.
- Processo tem apenas 3 estados: novo, executando e terminado.

um dos dois fica ocioso / CPU ou Disco

* SOs multiprogramados

- Vários programas/processos simultaneamente em memória.
- Permitem mecanismos de troca rápida de processo (especialmente se tiverem apenas 1 núcleo na CPU).
- ↳ pseudoparalelismo!
- Execução concorrente de tarefas.

↳ Subtipos de SOs multiprogramados

→ Sistemas de processamento em lotes (batch)

Consiste no envio de um lote/rotina de tarefas p/ serem executados de uma só vez. → programas!

Os jobs (scripts e/ou lotes de programa) dos usuários são submetidos em ordem sequencial p/ execução.

↳ Não há interação entre usuário e os jobs durante o período de execução

Exemplo de execução em lote: comando `make` do Linux.

Exemplo de execução em lote: comando `mkdirls` do `linux`.


Usuário apenas se submete!
NÃO interage.

Muito usados em máquinas Mainframe!

→ Sistemas multitarefa interativos

- Sistema responde na interação c/ o usuário a partir de diálogos.
- Pode, ou não, ser multiusuário.

* Estruturas de SO

→ Monolítica  acaba tendo menor latência!

- SO é um único módulo.
- SO é um conjunto de programas que executam sobre um hardware como se fossem um único programa.
- Programas de usuário invocam rotinas de SO.

Exemplos: VMX, `linux`, `Windows`.

→ Microkernel

- Busca tornar o núcleo do SO o menor possível, a fim de evitar erros que possam derrubar o SO.
- Busca atingir alta confiabilidade e modularização.
- Processos de usuário têm menos poder.

Exemplo: `Mimix`.

→ Comandos

- Busca criar um SO modular e hierárquico.
- Interface p/ facilitar interação entre os `fs` módulos.

Exemplos: `Multics`, `OpenVMS`.

→ Máquinas Virtuais

- Conciliam a presença de `fs` SOs em uma máquina a partir de um software chamado hipervisor, responsável por gerenciar a máquina.

de um software chamado hipervisor, responsável por gerenciar a máquina virtual.