

PP12 - Gestão do processador

Programa vs Processo

↳ conjunto de instruções que dexe a realização de uma det. tarefa por um computador.

↳ para a tarefa ser realizada, o programa tem que ser executado

processo - execução de um programa caracteriza-se por:

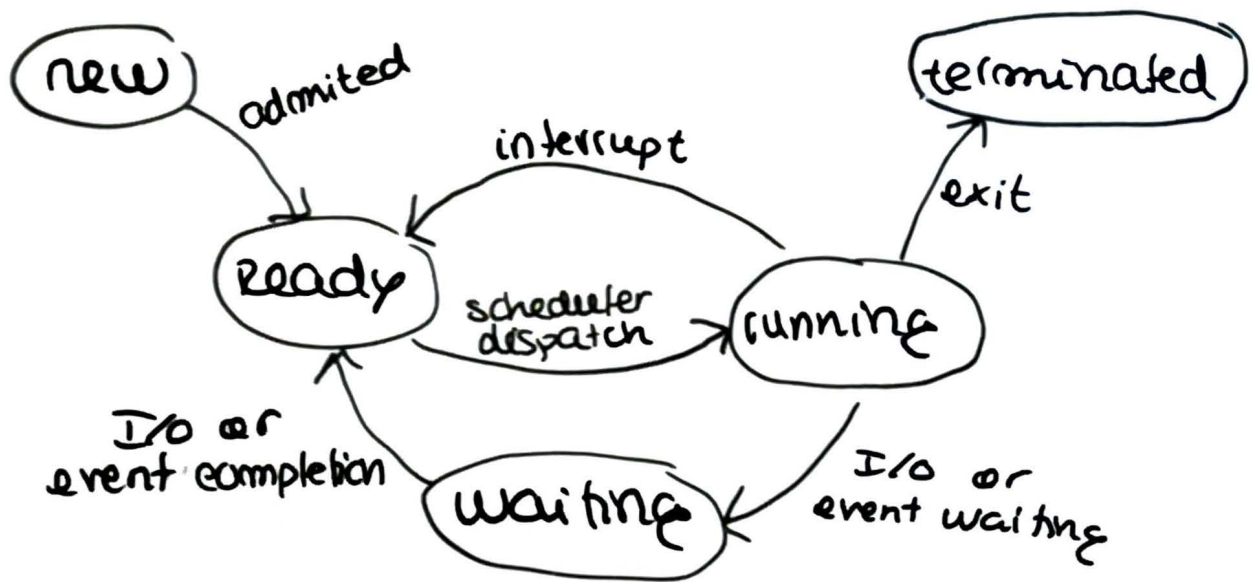
- espaço de endereçamento
- valor atual de todos os registos internos do processador
- os dados que estão a ser transferidos dos dispositivos de entrada para os de saída
- o seu estado de execução

Modelação de Processos

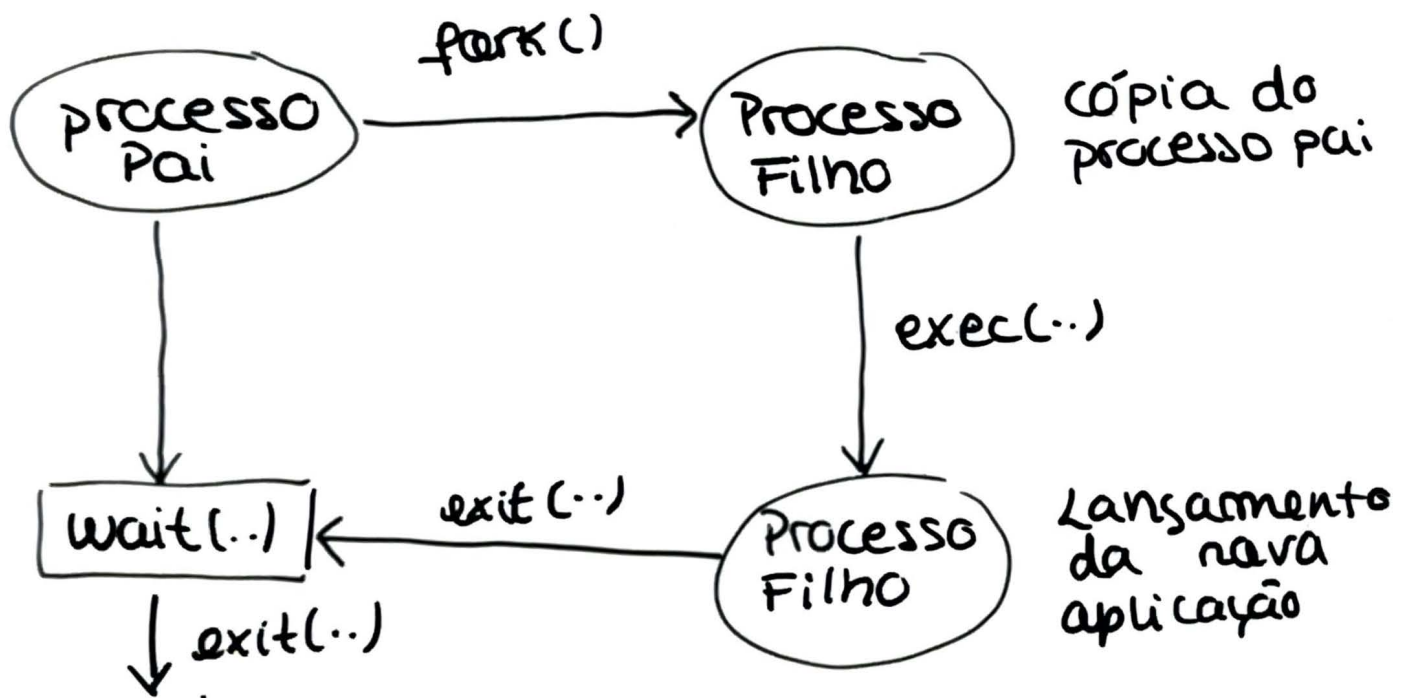
monoprocessador - n° de processadores virtuais ativos, é no máximo 1 em cada instante

multiprocessador - n° é no max = ao n° de processadores existentes

diagrama de estados



criação de um processo em Unix

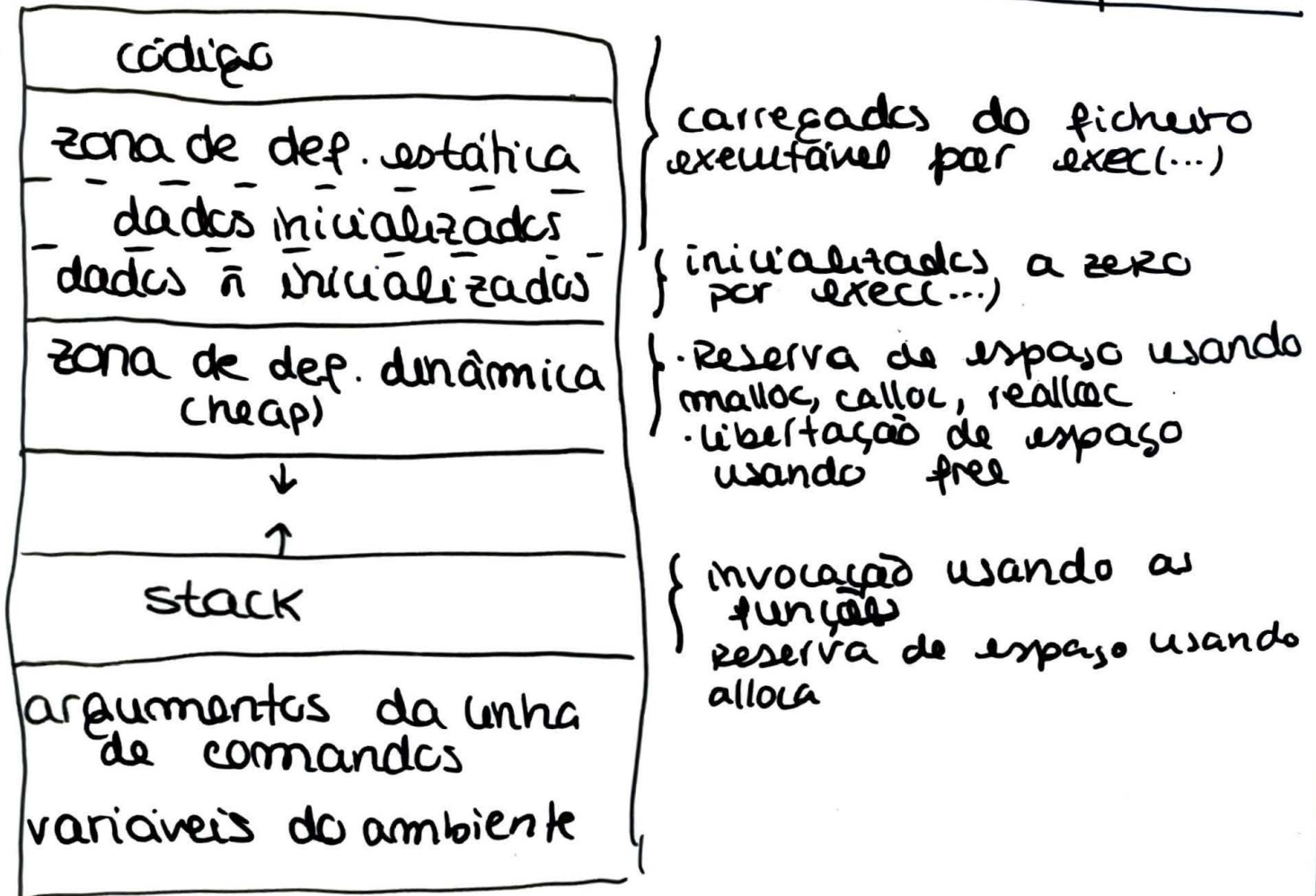


- processo pai cria processos filhos
- filhos podem criar novos processos
- partilha de recursos
 - pai e filho partilham recursos
 - filhos partilham um subconjunto dos recursos do pai
 - pai e filho \bar{n} partilham recursos

execução

- pai e filhos executam em paralelo
- pai espera que filho(s) terminem

Espaco de endereçamento de um processo



PCB - armazena a info. sobre os processos

- estado do processo
- program counter
- registos do CPU
- Tipo de escalonamento
- Info. sobre a memória do processo
- Info. sobre I/O
- Accounting

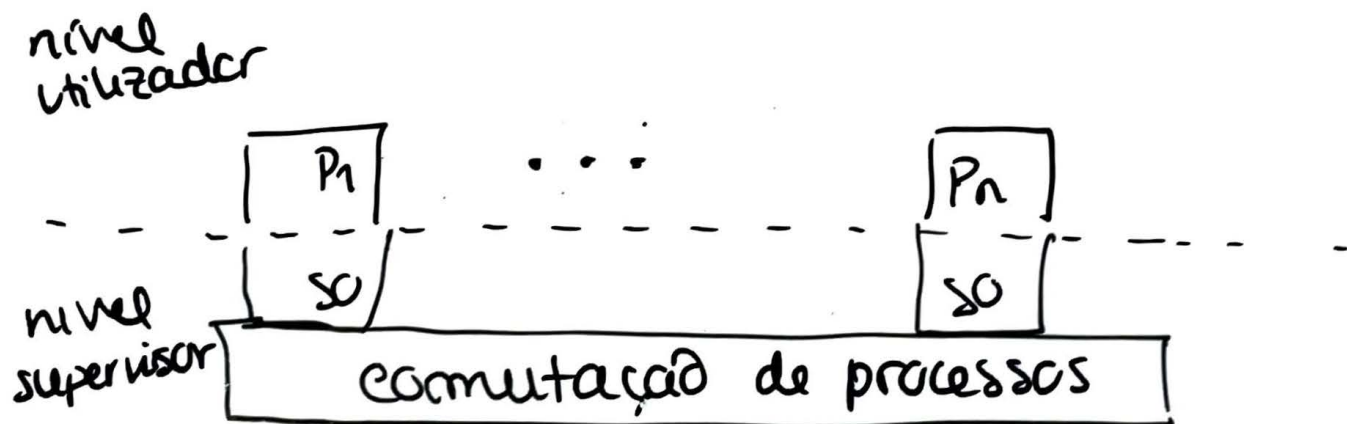
Comutação de processos

níveis de funcionamento dos processos

- nível supervisor
- nível utilizador

exceção - algo que interrompe a normal execução de instruções

- dispositivo externo (interrupção)
- execução de uma instrução ilegal, ou que conduz a um erro
- execução de uma instrução do tipo trap (interrupção por software)



Políticas de scheduling

- Non-preemptive scheduling (so tipo batch)
- Preemptive scheduling (so tipo interactivo)

Cr terios a satisfazer pelos algoritmos de scheduling

- justia
- previsibilidade
- throughput
- tempo de resposta
- tempo de turnaround
- deadlines
- efici ncia

Processes vs Threads

vantagens de um ambiente multithreaded

- > simplicidade na decomposi  o da tarefa e maior modularidade na implementa  o
- melhor gest  o de recursos do sistema computacional
- efici ncia e velocidade de execu  o

User level threads ⊕

↳ vis veis ao programador mas desconhecidas ao kernel

- uma biblioteca de threads no espa o do user normalmente usa user level threads

Kernel - level threads

o SO suporta e gerencia kernel level threads

⊕ ⊕ R pidas de criar e ⊕ f ceis de gerir que kernel level threads



Tipos de user e kernel threads

- mapeamento many-to-one
(user) (kernel)
- mapeamento one-to-one
↳ cada user thread para uma kernel thread correspondente
- mapeamento many-to-many
↳ múltiplas user threads para um $n \leq$ de kernel threads