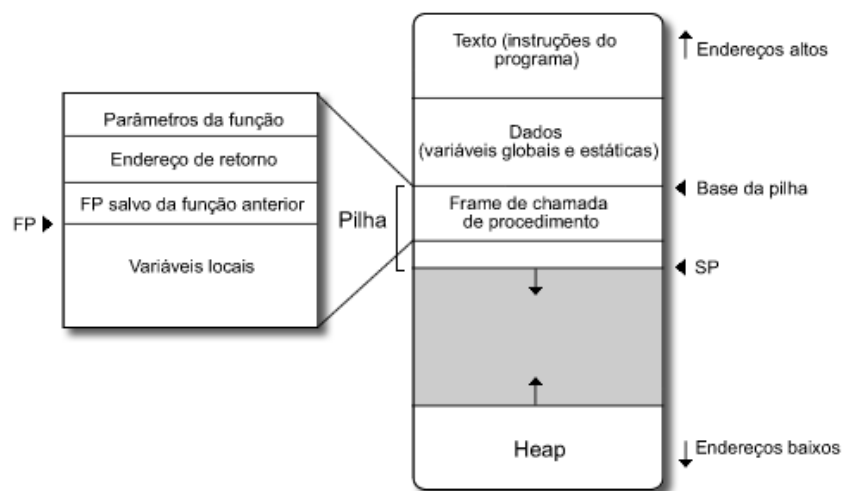


## Aula 6: Processos: Estados, PCB e Thread

Informalmente, um **processo** é um programa em execução. Um processo é mais do que o código do programa, conhecido algumas vezes como a **seção de texto**. O processo inclui também a atividade em curso, conforme representada pelo valor do **contador do programa** e pelos conteúdos dos registradores do processador. Além disso, um processo geralmente inclui a **pilha** do processo, que contém dados temporários (parâmetros de método, endereços de retorno e variáveis locais) e uma **seção de dados** que contém as variáveis globais.



**Figura 1: Informações de um processo.**

Enfatizamos que um programa por si só não é um processo; um programa é uma entidade *passiva*, como os conteúdos de um arquivo armazenado em disco, enquanto um processo é uma entidade *ativa* com um contador de programa especificando a próxima instrução a ser executada e com um conjunto de recursos associados.

### Estado do Processo

Um processo muda de estado durante sua execução. O **estado** de um processo é definido em parte pela sua atividade em curso. Cada processo pode estar em um dos seguintes estados:

- **Novo:** O processo está sendo criado.
- **Em execução:** As instruções estão sendo executadas.
- **Em espera:** O processo está esperando pela ocorrência de algum evento (como um término de entrada e saída ou recepção de um sinal).

- **Pronto:** O processo está esperando para ser designado a um processador.
- **Terminado:** O processo terminou sua execução.

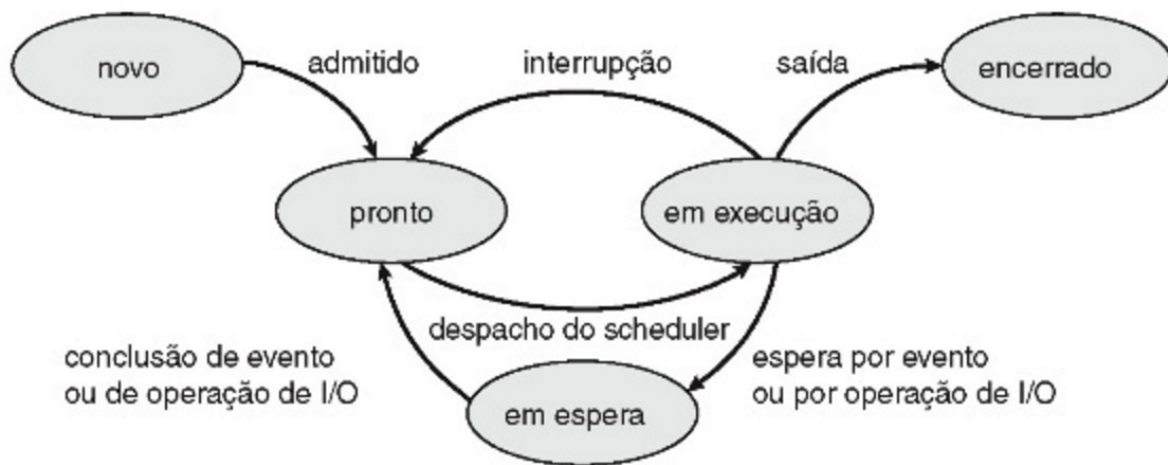
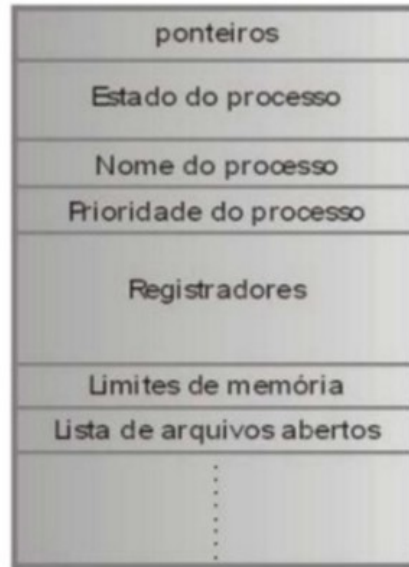


Figura 2: Diagrama de estado do processo.

## Bloco de Controle de Processo

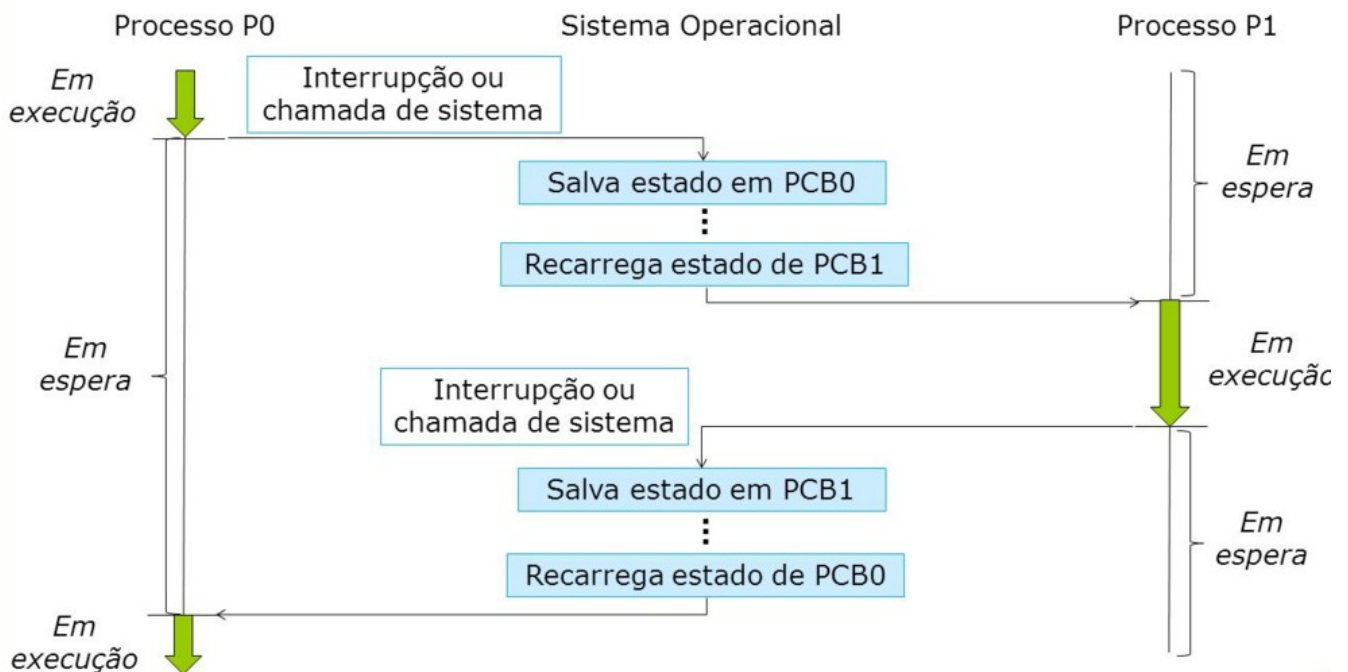
Cada processo é representado no sistema por um **bloco de controle de processo (PCB – process control block)**. Um PCB é mostrado na Figura 2. Ele contém muitas informações associadas ao processo específico, incluindo as seguintes:

- **Estado do processo:** O estado pode ser novo, pronto, em execução, em espera, etc.
- **Contador de programa:** O contador indica o endereço da próxima instrução a ser executada por este processo.
- **Registradores de CPU:** São acumuladores, registradores índices, ponteiros de pilhas, além de qualquer informação de código de condição. Quando ocorre uma interrupção, esta informação de estado deve ser salva juntamente com o contador do programa, para que o processo possa continuar corretamente mais tarde (Figura 3).
- **Informação de escalonamento de CPU:** Inclui prioridade do processo, ponteiros para filas de escalonamento e quaisquer outros parâmetros de escalonamento.
- **Informação de gerenciamento de memória:** Inclui registradores base e limite, tabelas de páginas e segmentos dependendo do sistema de memória utilizado.
- **Informação de contabilização:** Montante de tempo de CPU e de tempo real utilizado, limites de tempo, etc.
- **Informação de estado de entrada e saída:** Inclui a lista de dispositivos de entrada e saída alocados a este processo, uma lista de arquivos abertos, etc.



**Figura 3: Bloco de controle de processo (PCB).**

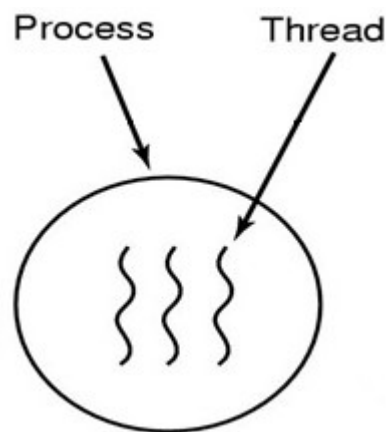
O PCB serve simplesmente como um repositório para qualquer informação que possa variar de um processo para outro.



**Figura 4: Diagrama mostrando a comutação da CPU de processo a processo.**

# Threads

O modelo de processo discutido até agora traz implícita a ideia de que um processo é um programa que está executando uma de suas linhas de ação (**threads**). Por exemplo, se um processo estiver executando um programa de processamento de textos, uma única linha de instruções do programa estará em ação. Esta única linha de controle permite ao processo executar somente uma tarefa de cada vez. Por exemplo, o usuário não poderia digitar caracteres e, ao mesmo tempo, executar o verificador de grafia dentro do mesmo processo.



**Figura 5: Relação entre processo e thread.**

## Exercícios

1. Descreva as ações tomadas por um sistema operacional para comutar o contexto entre processos.
2. Quais informações de escalonamento ficam armazenadas em um PCB de um processo?
3. Qual a relação existente entre Processo e Thread?
4. Qual local fica um processo que não foi executado, mas que está preparado para execução?
5. Liste os estados de um processo que irá apenas imprimir um arquivo.