

# Implementação de processos

Marcelo Johann

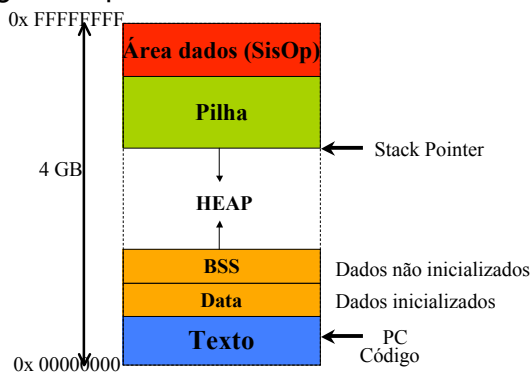
## Lembrando...

- Um processo = uma imagem dinâmica de um processo em execução.
- Um processo = um conjunto de recursos de HW alocados pelo Sis. Op. = uma unidade a escalonar.
- A imagem do processo na memória é constituída de vários segmentos.
- Um processo tem um ciclo de vida
  - Nasce, vive (ciclos de CPU/Ciclos de E/S), morre
- O Sis. Op. interage com o HW através de interrupções para reagir a erros ou eventos.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 2

## Imagem do processo Linux/Unix



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 3

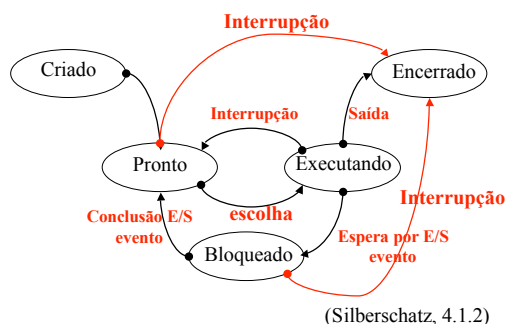
## Ciclo de vida – troca de estado (cont.)

- Ao executar, o processo pode querer fazer E/S:
  - O que acontece se o recurso de E/S está sendo ocupado?
  - É preciso de uma fila de processos *bloqueados*
- Após ter executado, o processo passa a ser *encerrado*.
- Em resumo:
  - 5 estados (criado, pronto, executando, bloqueado, encerrado)
  - Transições possíveis.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 4

## Diagrama de estados



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 5

## O que causa uma transição?

- Pronto -> executando
  - Algoritmo de escalonamento
- Executando -> pronto
  - Interrupção de tempo
  - Interrupção devida ao escalonador
  - Decisão espontânea (yield)
- Executando -> bloqueado
  - E/S – sincronização
- Bloqueado -> pronto
  - Interrupção
- Executando -> encerrado
  - Interrupção (Ctrl-C)
  - Término normal
- Bloqueado, pronto -> encerrado
  - Interrupção (Ctrl-C, kill)

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 6

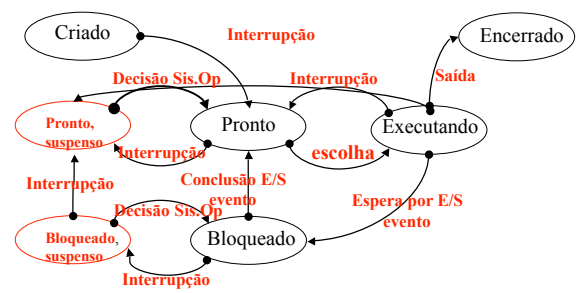
## Mais um estado: suspenso

- Dois problemas principais para gerar os recursos:
  - A CPU é muito mais rápida do que a memória;
  - A memória é de tamanho finito.
- Precisa-se, então, **poupar a memória**.
- Por isso:
  - Processos bloqueados que estão na memória podem ser transferidos para o disco (**swap**) até sua E/S ser acabada.
  - Processos prontos podem também ser descarregados para o disco.
- Chega-se a mais dois estados:
  - Bloqueado, suspenso.
  - Pronto, suspenso.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 7

## Transições com estado Suspenso



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 8

## Plano da aula

- Estrutura de dados para implementar um processo
  - Minix
- Implementação de interrupções sobre arquitetura Intel.
  - Troca de contexto

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 9

## Bloco Descritor de Processo

- A estrutura de dados que representa o processo no Sis. Op. é chamada "**Process Control Block**" (PCB)
- Contém as informações necessárias:
  - Registradores, memória, disco (arquivos)
  - Prioridade
  - Estado
  - Histórico (contabilidade)
  - Ponteiro para um outro PCB (lista encadeada)
- O Sis. Op. deve manter listas de processos
  - Listas encadeadas;
  - A estrutura PCB é usada para tal;
  - Mantém-se um ponteiro sobre o primeiro e/ou o último PCB.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 10

## Organização dos processos no Minix

- Existe 4 níveis de processos no Minix:
  - Nível 1: implementação baixo nível
    - Tratamento de interrupções
    - Escalonamento
    - Comunicação entre processos
  - Nível 2: processos que tratam de I/O (tarefas, device drivers)
  - Nível 3: gerenciadores de serviços
    - Memória, disco, rede...
  - Nível 4: processos do usuário.
    - Shell, compilador, editor (vi :o)...
- Núcleo = nível 1+nível 2.
  - Gerenciamento de recursos do HW
- Implementação das chamadas de sistema: nível 3.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 11

## Organização do código do Minix

- /usr/include/
  - Headers padrões
    - sys/ : headers adicionais
    - minix/ : headers usados pelo núcleo do Minix
    - Diretórios específicos ao hw.
- /usr/src/
  - Código fonte do Minix
    - kernel/ : o núcleo
    - mm/ : a memória
    - fs/ : o sistema de arquivos.
  - Código dos comandos do shell, programas de teste....

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 12

## Organização do código do Minix (2)

- /usr/include/minix
  - config.h: várias constantes necessárias nos outros arquivos.
    - Tamanho da tabela de processos...
  - type.h: alguns tipos básicos usados no Minix.
    - Mensagem.
  - Muitos outros...

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 13

## Código fonte do Minix

- /usr/src/kernel/
  - kernel.h: inclusão geral de outros arquivos.
  - proc.h: definição da struct proc e da tabela de processos
    - Cada entrada na tabela de processos é uma struct proc
    - P\_flags indica o estado do processo:
      - NO\_MAP, SENDING, RECEIVING, P\_STOP,...
  - rdy\_head e rdy\_tail ajudam manter as listas de processos.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 14

## Tarefas do Sis. Op. com o PCB - Minix

- Quando se cria um processo
  - Chama-se o construtor
  - Ele aloca espaço para um PCB
  - Inicializa (se for o caso) as listas
  - Ver servers/pm/forkexit.c: - do\_fork.c
- Durante o ciclo de vida
  - O PCB passa de uma lista para um outra;
  - Executam-se as instruções apontadas (PC)
  - Os campos do PCB vêm sendo alterados (vide uso dos recursos).
- Quando morre o processo
  - Chama-se o destrutor;
  - A memória é liberada;
  - Os recursos usados são liberados.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 15

## Interrupções do lado do núcleo do Minix

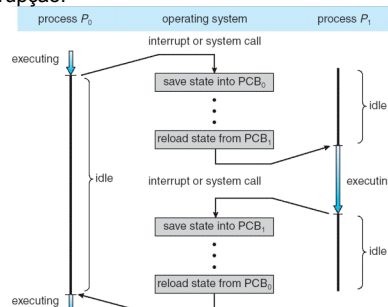
- O Byte de dado é usado como índice numa tabela de *handlers*.
  - Inicializada ao boot, ver main.c
- Cada handler implementa um tratamento a ser chamado.
- Como passar parâmetros?
  - Um ponteiro para uma área na memória,
  - Registradores,
  - Método mixto.
- Ao ser executado, um handler necessita efetuar uma troca de contexto.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 16

## Troca de contexto

- Um processo está em execução;
- Acontece uma interrupção!



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 17

## Linux

- Principal Sistema Operacional de código aberto de hoje.
  - Teve outros antes! (UNIX)
  - Têm outros hoje! (Minix, OpenSolaris, FreeBSD,...)
- Clone acadêmico do Unix.
  - Código aberto com **licença GPL** (= software de domínio público).

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2009/2

Aula 05 : Slide 18

## Histórico do Linux

- 1991 com o trabalho de Linus Torvalds sobre o kernel para i386 (kernel 0.01, maio 1991)
- Março de 94: kernel 1.0 com suporte de rede
  - Março de 95: kernel 1.2
    - suporte a novo HW (Sparc, Alpha).
- Junho de 96: versão 2.0
  - suporte a SMP, Sparc,
  - melhora na memória virtual e no sistema de arquivos,
  - *threads* no kernel,
  - módulos
- 2002 versão 2.4.x
  - melhora nos algoritmos de escalonamento das *threads*.
- Agora versão 2.6.17-8 (09/08/2006)

## Distribuições do Linux

- **Kernel & Distribuições:** em cima do kernel foram acrescentadas muitas aplicações para formar uma distribuição
  - Compiladores, Navegadores Web, Interpretadores de comandos (shells), interface gráfica (X)
  - Sistemas de gerenciamento/pre-compilação dos pacotes (RPM, APT)
- Distribuições:
  - Ubuntu, Debian, Gentoo...
  - RedHat, SuSe (Alemanha), Slackware, Mandrake, ...
  - Muitas outras!
- GUI vs. Linha de comando (xterm):
  - Gnome
  - KDE

## Referências sobre o Linux

- Na Web:
  - <http://www.kernel.org/> : o núcleo
  - [br-linux.org](http://br-linux.org/) : site generalista
  - <http://www.tldp.org/>
    - Ver os "Guides"!
- Jornais & Revistas:
  - [www.linuxjournal.com/](http://www.linuxjournal.com/)
  - <http://www.linux-magazine.com.br/>
- Livros editados:
  - Welsh & Kaufman, "Dominando o Linux", O'Reilly.
  - Bovet & Cesati, "Understanding the Linux Kernel" (O'Reilly 2005 – 3ª edição)

## Uso básico do Linux

- Se deslocar na árvore de diretórios:
  - cd, cd .., cd /, ...
- Editar um arquivo
  - vi, gedit, (x)emacs
- Listar o conteúdo de um diretório:
  - ls
- Deletar um arquivo/diretório
  - rm, rm -r
- Copiar um arquivo:
  - cp xxx.html yyy.htm
- Tar / gzip

## Dúvidas ou exemplos de C?

## Próxima aula...

- Laboratório Minix e criação de Processos
- Lab 101 Prédio 67