Sistemas Operacionais

CCP/SIF
UNISUL – Tubarão

Cassio Brodbeck Caporal
cassio{NOSPAM}ostec.com.br

Agenda

- Revisão;
- Estrutura de processos;
- *Process Control Block*;
- Estado de processos;
- Tipos de processos;
- Processos em primeiro e segundo plano;

Agenda

- Processos *CPU-bound* e *I/O-bound*;
- *Signals*;

Processos

- A CPU não tem conhecimento de qual programa está executando;
  - Função esta do sistema operacional;
- Um programa está sempre associado a um processo;
- Gerência de processos é função primordial em sistemas operacionais;

Processos

- Novamente...
  - Processos são executados concorrentemente, compartilhando:
    - Unidade Central de Processamento;
    - Memória principal;
    - Dispositivos de E/S.

Estrutura de Processos

- Resumidamente:
  - Processo é um programa em execução.
- Mas não é só isso: (novamente):
  - Um usuário tem idéia de que os recursos são exclusivos para a sessão;
  - Para que um programa possa voltar a ser executado, certas informações devem ser gravadas.

Estrutura de Processos

- Hmm …
  - Além de um programa em execução, faz parte de um processo as informações necessárias para que o mesmo possa voltar a executar futuramente;
  - Processo como um ambiente para a execução de programas: espaço de endereçamento, tempo de CPU, disco, etc.

Estrutura de Processos

- Elementos:
  - Contexto de hardware;
  - Contexto de software;
  - Espaço de endereçamento;
- Contexto de hardware:
  - “Local” de armazenamento de registradores gerais e específicos;

Estrutura de Processos

- Contexto de hardware:
  - O contexto de hardware é levado aos registradores quando um processo vai ser executado;
  - No momento em que outro processo toma a CPU, o sistema salva as informações no *contexto de hardware*;

Estrutura de Processos

- Contexto de hardware:
  - Mudança de contexto:
    - (1) Salva o conteúdo dos registradores do processo que será “paralizado”;
    - (2) Carrega a estrutura de *contexto de hardware* para os registradores (substituição de valores).

Estrutura de Processos

- Contexto de software:
  - Mantém informações como:
    - Número máximo de arquivos abertos;
    - Prioridades para execução;
    - Tamanho do *buffer* para E/S;
- Informações carregadas na criação do processo / Informações que *podem* ser alteradas durante a execução.

Estrutura de Processos

- Grupos de informações:
  - Identificação;
    - *Process Identification* (PID);
    - *Owner* (UID);
      - Segurança!
  - Quotas:
    - Especificação de limites: arquivos abertos, *buffer* para operações de E/S, máx. de operações de E/S pendentes, etc.

Estrutura de Processos

- Grupos de informações:
  - Privilégios:
    - Hierarquia de permissões com relação ao próprio processo, demais processos e restante do sistema operacional;
  - *Soft e hard limits*.

Estrutura de Processos

- Espaço de endereçamento:
  - Local de armazenamento de instruções e dados para execução;
  - Cada processo possui seu espaço:
    - É necessário a implementação de mecanismos que garantam o isolamento desta área por demais processos.

Estrutura de Processos

- *Process Control Block*:
  - Local de armazenamento de contexto de hardware, software e espaço de endereçamento;
  - Residente na memória principal, espaço exclusivo de uso do Sistema Operacional;

Estados de Processos

- Classificações:
  - *Running, ready* e *wait*;
  - Por que?
    - Novamente... Processos concorrem pelo uso dos recursos computacionais;
    - Não existe alocação exclusiva de CPU por um processo.

Estados de Processos

- Classificações:
  - *Running* (execução/executando):
    - Ocorre quando está sendo executado pela CPU;
    - Sistemas com múltiplos processadores podem ter processos ao mesmo tempo alocados em CPU's diferentes;

Estados de Processos

- Classificações:
  - *Ready* (pronto):
    - Esperando (na fila) para ser executado;
    - Algoritmos de escalonamento;
  - *Wait* (espera ou bloqueado):
    - Aguardando algum evento, como E/S, para ser executado.

Mudança de Estados

- Eventos voluntários;
- Eventos involuntários;
- Transicões de estados:
  - Pronto → execução;
  - Execução → espera;
  - Espera → pronto;
  - Execução → pronto;

Tipos de Processos

- Formas diferentes de implementar concorrência:
  - Processos independentes;
  - Subprocessos;
  - *Threads*;

Tipos de Processos

- Processos independentes:
  - Inexistência de hierarquia entre processos (pais e filhos);
  - PCB próprio para cada novo processo criado;
  - Maior consumo de recursos computacionais;

Tipos de Processos

- Subprocessos:
  - Estrutura hierarquizada;
  - Existência de processo pai e filhos;
  - Processos órfãos;
  - Existência de um PCB para cada subprocesso;
  - Alto custo para criação e finalização;
  - *fork()*.

Tipos de Processos

- *Threads*:
  - Criadas para aumentar desempenho, economizar recursos na criação de processos;
  - Inexistência de diversos subprocessos para gerenciar partes específicas do programa;
  - Compartilha somente *contexto de hardware*;
  - Intercomunicação extremamente rápida.

Processos *foreground*

- Processamento interativo (primeiro plano);
- Interface direta com o usuário para instruções de E/S;
- Análise de processos em *UNIX*;
  - Ferramentas:
    - ps, pstree, top, entre outros;

Processos *background*

- Processamento em segundo plano;
- Não há interação direta com usuário;
- Processos servidores (*daemons*);
- Análise de processos em *UNIX*;
  - Ferramentas:
    - ^Z, jobs, fg, bg;

CPU-*bound* e E/S-*bound*

- CPU-*bound*:
  - Utilização contínua de CPU (execução);
  - Pouca leitura e gravação, MUUITO cálculo;
- E/S-*bound*:
  - Maior parte do tempo fica no modo *espera*;
  - Grande utilização de operações de E/S;

Sinais

- Sinalização de processos da ocorrência de alguns eventos;
- Funcionamento semelhante a interrupções:
  - *Signal handler*;
- Sinais em UNIX:
  - SIGINT, SIGKILL, SIGTERM, SIGHUP, etc.