

06 – Contas de Usuários

06-Contas_de_usuarios

- **Tipos de contas:**
 - **Locais** (armazenadas no SAM, acesso restrito ao computador).
 - **Domínio** (armazenadas no Active Directory, acesso à rede).
 - **Internas** (administrator, guest).
 - **Diferenças:** Grupo de trabalho vs Domínio.
 - **Diretrizes:** convenções de nomes, políticas de senha, opções de restrição (horário, máquina, expiração).
 - **Atividade:** instalação do Active Directory.
-

Algoritmo do Banqueiro (Deadlock)

AlgoritmoBanqueiroDeadlock

- **Objetivo:** evitar alocação insegura de recursos.
 - **Estruturas:**
 - Available: recursos disponíveis.
 - Max: máximo que um processo pode requisitar.
 - Allocation: recursos atualmente alocados.
 - Need: recursos ainda necessários (Max – Allocation).
 - **Algoritmo de segurança:** verifica se existe sequência de execução segura.
 - **Pedido de recurso:** só é aceito se não ultrapassar Need, estiver em Available e manter o sistema em estado seguro.
 - **Exemplo:** 5 processos, 3 tipos de recursos, sequência segura <P1, P3, P4, P2, P0>.
-

Exercício – Banqueiro para Deadlocks

Exercicio - banqueiro para Dead...

- Aplicação prática do algoritmo do banqueiro.
 - Questões:
 - a) Verificar se o sistema está em estado seguro.
 - b) Pedido de P1 = (0,4,2,0): analisar se pode ser atendido.
 - c) Pedido de P2 = (1,0,0,2): verificar se pode ser concedido, e em que ordem de execução.
-

Lista Extra de Exercícios – Concorrência de Processos

Lista de Exercícios Extra - Con...

- Problemas clássicos de sincronização:
 1. Impressão em ordem usando semáforos.
 2. Grafo de precedência entre processos.
 3. **Pombo correio** (20 mensagens).
 4. **Jantar dos Canibais** (cozinheiro repõe comida).
 5. **Filme sobre Hoare** (lotação fixa de 10 pessoas).
 6. **Leitores e Escritores** (acesso concorrente a BD).
 7. **Barbeiro Dorminhoco**.
-

SO1 – Parte 4: Concorrência

SO1-Parte4-Concorrencia

- **Conceitos:** processos concorrentes, comunicação interprocessos, exclusão mútua.
- **Problemas:** condição de corrida, regiões críticas.
- **Soluções:**
 - Hardware: inibição de interrupções, instrução TSL.
 - Software: variáveis de bloqueio, alternância estrita, algoritmos de Dekker e Peterson.
 - Bloqueio: semáforos, monitores.

- **Exemplos:** produtor-consumidor, spool de impressão.
-

■ SO1 – Parte 5: Deadlocks (resumida)

SO1-Parte5-Deadlocks-resumida

- **Definição:** conjunto de processos esperando recursos uns dos outros indefinidamente.
 - **Condições de Coffman:** exclusão mútua, posse e espera, não-preempção, espera circular.
 - **Representação:** grafo de alocação de recursos.
 - **Métodos de tratamento:**
 - **Prevenção** (quebra de condições de Coffman).
 - **Impedimento** (algoritmo do banqueiro, estados seguros).
 - **Detecção e recuperação.**
 - **Ignorar** (algoritmo do avestruz, usado em UNIX).
-

■ SO1 – Parte 6: Threads (completa)

SO1-Parte6-Threads-completa

- **Conceito:** unidade leve de execução dentro de um processo.
- **Vantagens:** criação rápida, comunicação simples, menos recursos que processos.
- **Modelos:**
 - **ULT (M:1)** – várias threads de usuário sobre uma thread do kernel.
 - **KLT (1:1)** – cada thread de usuário mapeada em uma do kernel.
 - **Híbrido (M:n)** – múltiplas threads de usuário sobre múltiplas do kernel.
- **API e comandos:** criação, join, yield, exit.
- **Exemplos:** processador de texto multithread, servidor web.

- **Java Threads:** criação (herança ou interface Runnable), sincronização com synchronized, wait, signal.
- **Monitores:** controle de exclusão mútua e variáveis de condição.