	COLÉGIO ESTADUAL PROTÁSIO ALVES ENSINO HÍBRIDO - 2021	
	PROFESSORA: Maria Helena - maria-hsilva368@educar.rs.gov.br DISCIPLINA: Sistemas Operacionais CONTEÚDO: Gerência de Memória	ATIVIDADE:05
	ALUNO: _____ TURMA: _____	DATA: 04/05/2021
	OBSERVAÇÕES: Gerenciamento de Memória - Memória Virtual (MV)	

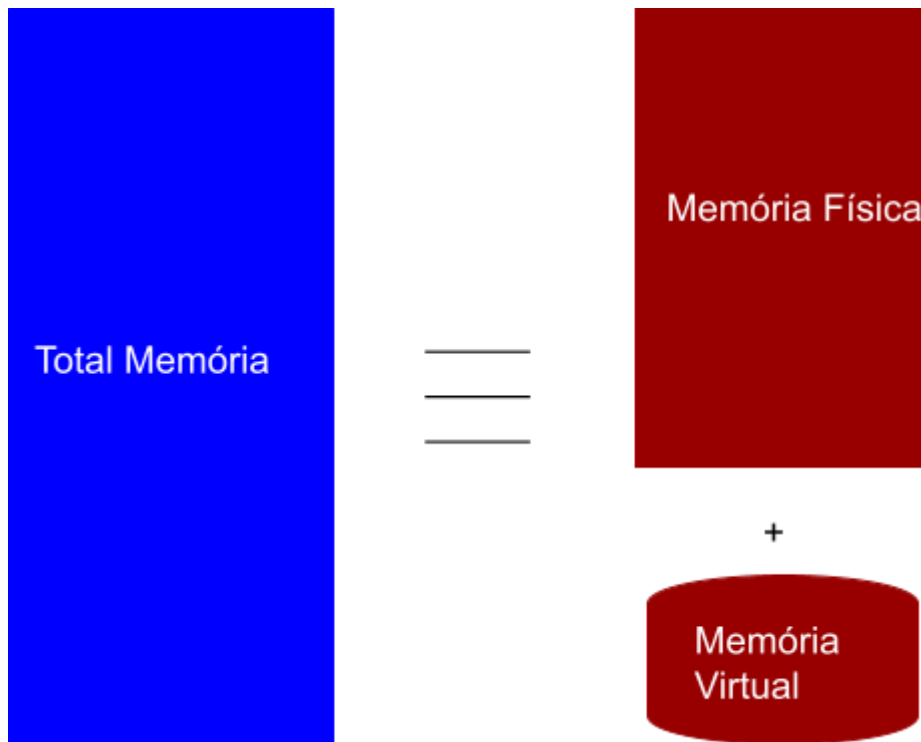
Gerenciamento de Memória - Memória Virtual (MV)

objetivo: Compreender as técnicas aplicadas pelo SÓ na implementação Memória Virtual (MV). Conhecer os fundamentos dos algoritmos utilizados nesse processo e a forma como há interação das memórias secundárias e primárias.

Memória Virtual

O conceito de relocação de memória possibilitou o desenvolvimento de um meio mais otimizado de utilização de memória, chamada de memória virtual. O conceito de memória virtual está fundamentado em desvincular o endereçamento feito pelo programa dos endereços físicos da memória principal. Assim, os programas e suas estruturas de dados deixam de estar limitados ao tamanho da memória física disponível.

O termo memória virtual é normalmente associado com a habilidade de um sistema endereçar muito mais memória do que a fisicamente disponível. Este conceito surgiu em 1960 no computador Atlas, construído pela Universidade de Manchester (Inglaterra), embora sua utilização mais ampla tenha acontecido recentemente.



A memória virtual de um sistema é, de fato, o(s) arquivo(s) de troca ou swap file(s) gravado(s) no disco rígido. Portanto, a memória total de um sistema, que possui memória virtual, é a soma de sua memória física, de tamanho fixo, com a memória virtual. O tamanho da memória virtual, chamado de arquivo de paginação no Windows XP, é definido, basicamente, pelo menor valor entre os seguintes:

- Capacidade de endereçamento do processador.
- Capacidade de administração de endereços do SO.
- Capacidade de armazenamento dos dispositivos de armazenamento secundário (unidades de disco).

Espaço de Endereçamento Virtual

Um programa no ambiente de memória virtual não faz referência a endereços físicos de memória (endereços reais), mas apenas a endereços virtuais. No momento da execução de uma instrução, o endereço virtual é traduzido para um endereço físico, pois o processador acessa apenas posições da memória principal.

O mecanismo de tradução do endereço virtual para endereço físico é denominado mapeamento. Esse mecanismo, nos sistemas atuais, é feito pelo hardware, com o auxílio do SO, e ele traduz um endereço localizado no espaço de endereçamento virtual para um endereço físico de memória, pois o programa executado em seu contexto precisa estar no espaço de endereçamento real para poder ser referenciado ou executado. Portanto, um programa não precisa estar necessariamente contíguo na memória real para ser executado.

O mecanismo de tradução se encarrega de manter tabelas de mapeamento exclusivas para cada processo, relacionando os endereços virtuais do processo às suas posições na memória física.

A memória virtual pode ser implementada por meio do uso dos mecanismos de paginação e segmentação. Atualmente, a paginação é o mecanismo mais utilizado na implementação de memória virtual, já a segmentação é uma alternativa menos utilizada, porém, é mais adequada do ponto de vista de programação. Há sistemas que utilizam ambas as técnicas.

Paginação

A paginação é uma técnica de gerenciamento de memória que utiliza o conceito de memória virtual, ou seja, a quantidade de endereçamento é maior do que o tamanho da memória fisicamente presente no sistema.

Desta forma, o endereçamento global ou espaço de endereçamento virtual é dividido em pequenos blocos iguais conhecidos como páginas virtuais.

Cada página possui um número que a identifica, e a memória física é dividida em blocos iguais, com o mesmo tamanho das páginas virtuais, chamado de molduras de páginas. As molduras de páginas são identificadas por um número e correspondem a determinada área da memória física. Todas as molduras de páginas são identificadas a partir do número zero.

Os endereços gerados por um programa em execução são chamados de endereços virtuais e formam o já mencionado espaço de endereçamento virtual. Seja o caso de um programa executar a seguinte instrução de certa linguagem de programação: MOV REG, 2060. Essa instrução, desta linguagem, diz que o conteúdo do endereço de memória 2060 deve ser copiado para o registrador REG.

O endereço 2060, gerado pelo programa, é, portanto, um endereço virtual. Em computadores sem o mecanismo de memória virtual, o endereço virtual é colocado diretamente no barramento da memória, e isto causará o acesso (leitura ou escrita) da palavra de memória física com o mesmo endereço. No caso de existir memória virtual, o endereço virtual vai para a MMU (Memory Management Unit – Unidade de Gerenciamento de Memória), um CI ou uma coleção de CIs (Circuitos Integrados) que faz o mapeamento dos endereços virtuais em endereços físicos.

Gerenciamento de processador - semana 11 a 14 de maio

objetivo: Compreender as técnicas e critérios de escalonamento aplicados ao gerenciamento do processador aplicados pelo SO. Definir os objetivos utilizados em tipos diferentes de SO utilizados pelo gerência do processador levando em consideração os tipos de sistemas operacionais.

Conteúdo

Introdução

critério na política de escalonamento

tipos de escalonamento

vocabulário

tipos de SO

Introdução

A gerência do processador tornou-se uma técnica importante do sistema operacional à medida que os sistemas multiprogramáveis começaram a ser mais utilizados, onde múltiplos processos são compartilhados pela CPU. Esta gerência e a multiprogramação esta estruturada com base na política de escalonamento. O gerenciamento trata o processador como um recurso compartilhado entre usuários externos ao processador e processos internos a ele. O resultado deste tratamento é o que chamamos de gerenciamento do processador e consiste em dois tipos de escalonamentos: o escalonamento a longo prazo da carga externa e o escalonamento a curto prazo dos processos internos. No ambiente multithread, cada processo pode responder a várias solicitações

concorrentemente ou mesmo simultaneamente, caso haja mais de um processador. A grande vantagem do uso de threads é a possibilidade de minimizar a alocação de recursos do sistema, além de diminuir o overhead na criação, troca e eliminação de processos. Threads compartilham o processador da mesma maneira que processos e passam pelas mesmas mudanças de estado. Para permitir a troca de contexto entre os diversos threads, cada thread possui seu próprio contexto de hardware, com o conteúdo dos registradores gerais, PC e SP. Quando um thread está sendo executado, seu contexto de hardware está armazenado nos registradores do processador. Throughput: significa o número de processos executados em um determinado intervalo de tempo. Quanto maior o throughput, maior o número de tarefas executadas em função do tempo. A maximização do throughput é desejada na maioria dos sistemas.

Critérios e tipos de escalonamento

O objetivo do escalonamento é maximizar a utilização da CPU via multiprogramação. O escalonamento de processador ocorre quando existe uma política do sistema para escolher quais processos executar. A política de escalonamento deve atender algumas premissas ligadas a desempenho e maximização do número de processos executados, reduzir o tempo de latência (tempo de espera antes da execução), evitar antecipação indefinida de processos e aperfeiçoar o uso do(s) processador (es).

O critério ajuda a determinar qual processo será usado em qual processador. E de acordo com a seleção dos critérios compõe-se a política de escalonamento. Essa política é uma das diferenças entre os sistemas operacionais existentes no mercado, pois cada sistema operacional possui a sua política adequada a suas características. Apesar de cada sistema operacional adotar seus critérios, podemos citar alguns dos principais:

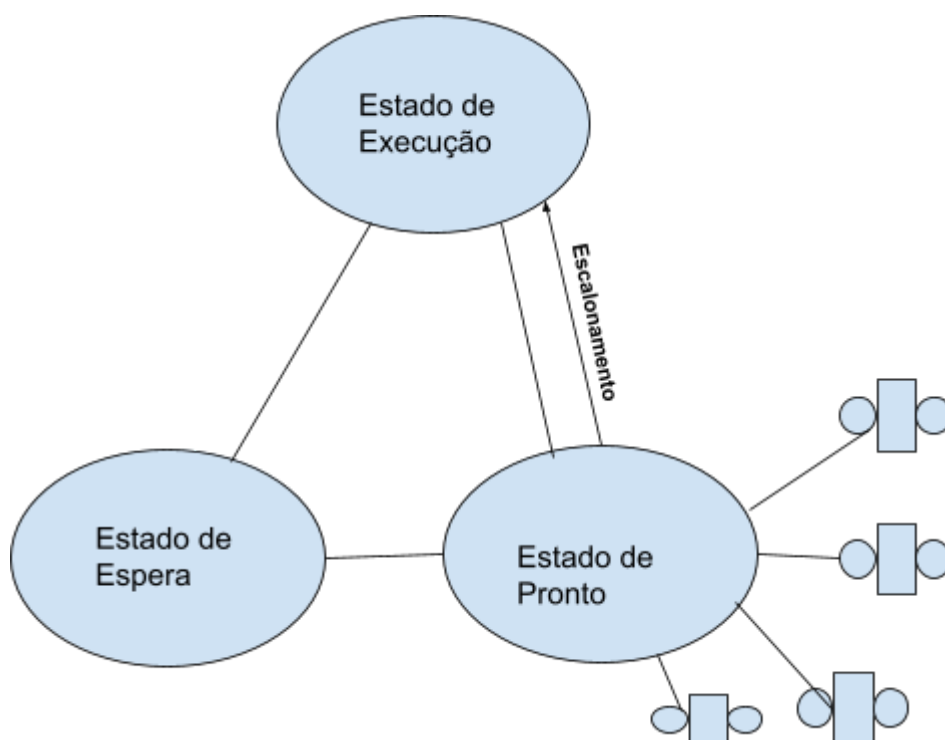
- Todos os processos são tratados igualmente, não ficando processo sem tratamento;
- Deve maximizar seu desempenho com maior número possível de processos por unidade de tempo; Atender os usuários em tempos de respostas aceitáveis;
- Previsibilidade é fundamental independentemente do sistema; Mitigar recursos perdidos; Balancear o uso de recursos;
- Ocupar o máximo o processador;

- Evitar espera indefinida (starvation), e principalmente o deadlock;
- Priorizar os processos de forma a identificar os de maiores prioridades;

Funções básicas da Política de Escalonamento de Processos A política de escalonamento de um sistema operacional tem diversas funções básicas:

- Manter o processador ocupado a maior parte do tempo;
- Balancear o uso da UCP entre os processos;
- Privilegiar a execução de aplicações críticas;
- Maximizar o throughput do sistema;
- Oferecer tempos de resposta razoáveis;

Cada sistema operacional possui a sua própria política de escalonamento adequada ao seu propósito.



A rotina Scheduler

A rotina Scheduler tem como função implementar os critérios da política de escalonamento. Esta rotina também é chamada escalonador. Todo o compartilhamento do processador depende desta rotina.

A rotina dispatcher

A rotina dispatcher tem como função realizar a troca de contexto dos processos após o escalonador determinar qual processo deve fazer uso do processador. O tempo gasto com a troca de contexto do processo é chamado latência do dispatcher.

Em ambientes que implementam apenas processos (seu suporte a threads) o escalonamento é realizado com base nos processos prontos para execução.

Em sistemas que implementam threads, o escalonamento é realizado considerando os threads no estado de pronto. Neste caso, podemos considerar o processo como sendo a unidade de alocação de recursos, enquanto o thread é a unidade de escalonamento.

Os critérios de escalonamento

A política de escalonamento depende das características do sistema operacional.

Em sistemas de tempo compartilhado o escalonamento trata todos os processos de forma igual, evitando que um processo fique indefinidamente esperando pelo uso do processador.

Em **sistemas de tempo real** o escalonamento sempre prioriza a execução de processos críticos , com alta prioridade, em detrimento de outros processos.

Os principais critérios que são considerados em uma política de escalonamento são os seguintes;

Utilização do processador Quase em todos os sistemas é esperado que o processador esteja a maior parte do tempo ocupado.

Throughput

Throughput representa o número de processos executados em um determinado intervalo de tempo.

Tempo de Processador / Tempo de CPU

É o tempo que um processo leva no estado de execução durante seu processamento. A política de escalonamento não influencia este tempo que depende apenas do programa executado e do volume de dados processados.

Tempo de Espera

Tempo de espera é o tempo total que um processo permanece na fila de pronto durante seu processamento, aguardando para ser executado.

Tempo de Turnaround

Tempo de turnaround é o tempo que um processo leva desde a sua criação até seu término.

Tempo de Resposta

Tempo de resposta é o tempo decorrido entre uma requisição ao sistema e o instante em que a resposta é exibida.

Em todos os casos, uma política de escalonamento sempre busca otimizar o uso do processador e do throughput.

Tipos de escalonamento

Escalonamento Não-Preemptivos e Preemptivos

As políticas de escalonamento podem ser classificadas segundo a possibilidade de o sistema operacional interromper um processo em execução e trocá-lo por outro processo, atividade essa conhecida por preempção.

Sistemas operacionais que implementam escalonamento com preempção são mais complexos, contudo permitem políticas de escalonamento mais flexíveis.

O escalonamento Não-Preemptivo foi o primeiro tipo de escalonamento implementado nos sistemas operacionais multiprogramados, onde predominava processamento batch.

No escalonamento Não-Preemptivo quando um processo em execução, um batch, nenhum evento externo pode provocar a perda do uso do processador. Ou seja, um processo em execução somente sai do estado em execução quando termina seu processamento ou quando ocorre um erro devido por algum código do próprio processo.

No Escalonamento Preemptivo o sistema operacional pode interromper um processo em execução e passá-lo para o estado de pronto, para alocar outro processo na UCP.

Com o uso da Preempção é possível ao sistema operacional priorizar a execução de processos, como no caso de aplicações de tempo real, onde o fator tempo é crítico.

Outro benefício é a possibilidade de implementar políticas de escalonamento que compartilhem o processador de uma maneira mais uniforme, distribuindo de forma balanceada o uso da UCP pelos processos.

Escalonamento first-in-first-out (FIFO)

Neste escalonamento o processo que chegar primeiro ao estado de pronto é selecionado para execução. Os processos em estado pronto são colocados em uma fila de pronto. Quando o processo

de execução é colocado no estado de espera, o primeiro processo da fila pronto é selecionado para execução.

Escalonamento Shortest-Job_First (SJF)

Neste escalonamento o processo que tiver o menor tempo de processador é selecionado para execução, ou seja, o processo que necessitar de menor tempo de UCP para terminar seu processamento é selecionado para execução.

Escalonamento Cooperativo

Neste escalonamento o processo em execução na UCP pode liberar o processador, retornando à fila de pronto e possibilitando que um outro processo seja escalonado. Desta forma, há uma cooperação para uma melhor distribuição de uso do processador.

As primeiras versões do sistema operacional Windows utilizavam este tipo de escalonamento.

Escalonamento Circular

Neste escalonamento do tipo preemptivo, projetado especialmente para sistemas de tempo compartilhado. Este escalonamento é parecido com o escalonamento FIFO porem quando um processo passa para o estado de execução existe um tempo-limite para o uso contínuo o processador, uma fatia de tempo para cada processo, chamada time-slice.