

Sistemas operacionais: Conceitos e Mecanismos

Capítulo 4: O conceito de tarefa

Integrantes do grupo: Jefferson Botitano e Leonardo Ludvig
Professor: Arliones Stevert Hoeller Junior
Disciplina: Sistemas operacionais

O conceito da tarefa

A tarefa em um sistema operacional tem o intuito de realizar o um fluxo sequencial de instruções sendo esse conjunto de instruções denominado programa que possui uma finalidade específica, como por exemplo a realização de cálculos, formatação de disco, editar algum gráfico entre outros. O procedimento de realizar esta sequência de instruções que é compreendido apenas pela máquina resultado da compilação do programa é denominado tarefa.

A gerência de tarefas

O processador de um computador necessita de executar todas as tarefas submetidas pelos usuários, para lidar com essas tarefas com distintas formas de execução e comportamento, o sistema operacional necessita de uma forma de organizar essas tarefas para decidir como e em qual ordem deve executar elas.

- Sistemas monotarefa: Primeira forma de executar tarefas, realizada manualmente pela troca de discos de programa, feito pelos operadores humanos.
- O monitor de sistema: Precursor dos sistemas operacionais, era um gerenciador de tarefas que executava uma fila de tarefas armazenadas em um disco de forma sequencial.
- Sistemas multitarefas: Para contornar o problema do sistema monitor onde o processador fica ocioso no tempo em que o sistema busca os recursos necessários para executar a tarefa, os sistemas multitarefas tem a capacidade de suspender uma tarefa que não está pronta para ser executada enquanto ocupa o processador com uma tarefa pronta para ser executada.
- Sistemas de tempo compartilhado: Para solucionar o problema de programas que executam em loop e ocupam a posição de tarefa em execução para sempre no sistema foi necessário implementar um temporizador que suspende a tarefa caso ela execute por um tempo determinado.

Ciclo de vida das tarefas

As tarefas podem ter o seu estado atual definido através do denominado ciclo de vida das tarefas, onde cada estado de uma tarefa recebe um nome e comportamento definido, como:

- **Nova:** Quando a tarefa é criada, seu código e suas bibliotecas estão sendo carregadas na memória e as estruturas de dados do núcleo sendo atualizadas para sua execução.
- **Pronta:** A tarefa está em memória esperando o processador para ser executada, tarefas prontas são organizadas em filas.
- **Executando:** Estado onde a tarefa está sendo executada pelo processador dedicado a ela.
- **Suspensa:** A tarefa está em um estado de espera devido a falta de dados externos ainda não fornecidos ou recursos não disponíveis ou até mesmo por ter outra tarefa em execução.
- **Terminada:** O processamento da tarefa foi encerrado e ela é finalizada e removida da memória do sistema.

Entretanto essa classificação ciclo de vida das tarefas não tem muito sentido nos sistemas atuais baseados em memória paginada onde processos podem ser executados mesmo parcialmente carregados na memória.

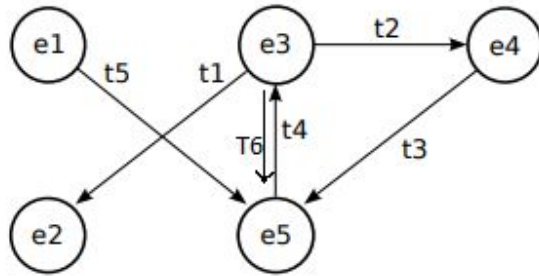
O que significa time sharing e qual sua importância em um sistema operacional?

Seu significado literal é tempo compartilhado, sendo uma solução criada no desenvolvimento dos sistemas operacionais, onde uma tarefa que ficasse em execução e não conseguisse finalizar a sua execução por algum motivo ficava permanente em execução ocupando o processador indeterminadamente. A solução encontrada para isso foi introduzir um tempo de processamento limite para cada tarefa denominada “quantum”, esgotado o seu quantum a tarefa volta para a fila de execução para esperar seu momento de ser processada novamente.

Como e com base em que critérios é escolhida a duração de um quantum de processamento?

A duração de um quantum é definida pelo processador ao receber uma tarefa, ele ajusta um contador que vai decrementado a cada tempo X tempo já predeterminado chamado de tick, que nada mais é que um temporizador programado para gerar interrupções em um intervalo regular de tempo. Após o contador de ticks de uma tarefa chegar a zero seu quantum acaba, fazendo o processador retirar a força os recursos da tarefa.

Considerando o diagrama de estados dos processos apresentando a figura a seguir, complete o diagrama com a transição de estado que está faltando(t6) e apresente o significado de cada um dos estados e transições.



- e1: Novo
- e2: Terminado
- e3: executando
- e4: Suspenso
- e5: Pronto

- t1: execução finalizada
- t2: esperando por evento ou tempo.
- t3: espera termina, pode continuar
- t4: Recebe processador
- t5: Pronto para executar
- t6: preempção de tempo

Indique se cada uma das transições de estado de tarefas a seguir definidas é possível ou não. Se a transição for possível, dê um exemplo de situação na qual ela ocorre (*N*:Nova, *P*:Pronta, *E*:Executando, *S*:Suspensa, *T*:Terminada):

- a) $E \rightarrow P$: Esgota o quantum da tarefa.
- b) $S \rightarrow T$: Não ocorre.
- c) $E \rightarrow S$: Tarefa está em estado de aguardo por um recurso ou evento externo.
- d) $E \rightarrow T$: Tarefa encerra execução.
- e) $S \rightarrow E$: Não ocorre.
- f) $N \rightarrow S$: Não ocorre.
- g) $P \rightarrow N$: Não ocorre.
- h) $P \rightarrow S$: Não ocorre.

Relacione as afirmações abaixo aos respectivos estados no ciclo das tarefas (*N*:Nova, *P*:Pronta, *E*:Executando, *S*:Suspensa, *T*:Terminada):

- [N] O código da tarefa está sendo carregado.
- [P] A tarefas são ordenadas por prioridades.
- [E] A tarefa sai deste estado ao solicitar uma operação de entrada/saída.
- [T] Os recursos usados pela tarefa são devolvidos ao sistema.
- [P] A tarefa vai a este estado ao terminar seu quantum.
- [P] A tarefa só precisa do processador para poder executar.
- [S] O acesso a um semáforo em uso pode levar a tarefa a este estado.
- [E] A tarefa pode criar novas tarefas.
- [E] Há uma tarefa neste estado para cada processador do sistema.
- [B] A tarefa aguarda a ocorrência de um evento externo.