



TP01 - Sistema Operacional

Aluno: Nathann Zini dos Reis

Matrícula: 19.2.4007

Q.1 - Defina as gerações de computadores presentes no livro do Tanenbaum, explicando

TAMBEM o que é uma tarefa, um sistema operacional multitarefa, a diferença para a definição de time-sharing (são parecidos, mas o foco é diferente), time-sharing e spooling.

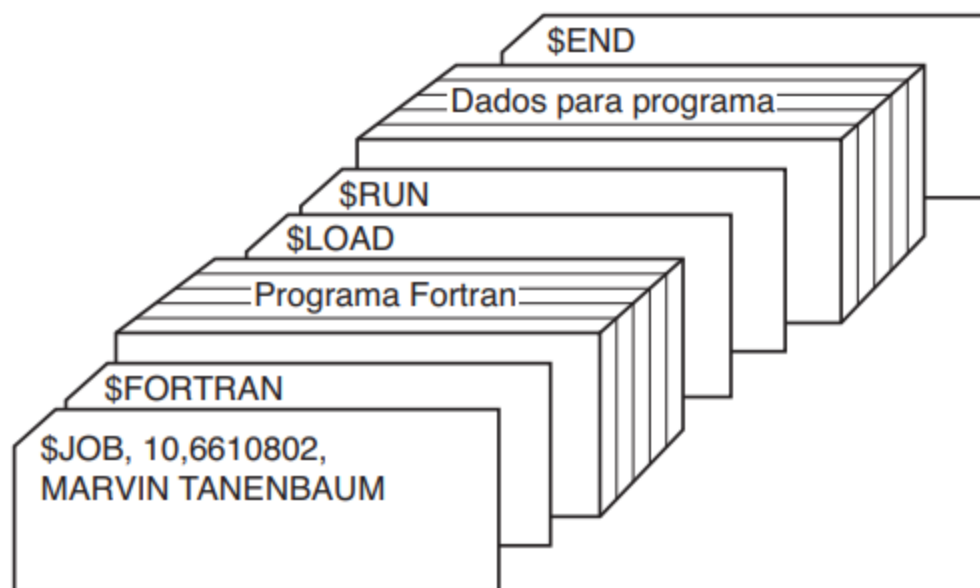
- 1º Geração: A Primeira geração de computadores compreende o tempo de 1945 a 1955 e ficou conhecida como a geração Válvulas, pois um computador dessa geração chegava a possuir até 20 mil válvulas para que pudesse funcionar. É importante pontuar que os computadores desta geração apenas realizavam cálculos numéricos diretos, como determinar tabelas de senos e cossenos e logaritmos. Ainda não havia sido inventado o sistema operacional até então.
- 2º Geração: A segunda geração de computadores compreende o tempo de 1955 a 1965 e foi marcada pela introdução dos transistores e sistemas em lote (batch). Com os transistores, os computadores começaram a ser comercializado, já que agora então eles tinham garantia de funcionamento por tempo suficiente para executar algum trabalho útil, pelo menos. Entretanto, ainda eram computadores de grande porte, portanto eram extremamente caros. Logo, ele não era comercializado para fins domésticos, mas para grandes empresas e instituições que tinham condições de bancar. As atividades dos computadores dessa geração eram separadas pelo programador, que ficava responsável de criar os cartões e pelo operador que ficava responsável por operar a máquina.

A realização dessa atividade era muito custosa em questão de tempo, então foi pensado em uma solução para tentar minimizar esse custo, que foi a criação do

sistema de lotes, que basicamente armazenava agora não apenas 1 tarefa por vez, mas um conjunto de tarefas que eram acumulados em uma fita magnética por aproximadamente uma hora. Para diminuir o custo e melhorar o tempo de execução, as tarefas eram levadas para um computador (1401) inferir em relação a execução do programa, mas mais eficaz para a leitura e para gerar a saída das tarefas, e também é um computador mais barato e de menor porte. As tarefas eram levadas para esse computador e era gerado uma fita magnética com o conjunto de tarefas que então eram levadas para o computador de grande porte para executar uma tarefa por vez e, ao invés de imprimir a saída, a saída era gravada em uma outra fita magnética de saída, que então era levada para um outro computador 1401 para que pudesse ser gerada ou impressa a saída das tarefas que foram executadas daquela fita magnética de entrada.

Uma tarefa é a execução de um fluxo sequencial de instruções, que foram construídas para atender uma finalidade específica, como realizar um cálculo, execução de um software e etc.

Esse conjunto de instruções está representado na imagem a baixo



em que o `&JOB` representa tempo máximo de processamento em minutos, o número da conta a ser debitada e o número do programador. `&FORTRAN` mandava o

SO carregar o compilador Fortran. &LOAD que era responsável por ordenar ao SO que carregasse o programa recém compilado. &RUN que fazia com que o SO executasse o programa com os dados fornecidos em seguidas e o &END que indica o término da execução daquele conjunto de instruções, ou tarefa.

- 3º Geração: A terceira geração contempla o período de 1965 a 1980 e foi marcada pela adesão dos Circuitos integrados e pela multiprogramação/multitarefas. A diferença entre os dois computadores, o mais simples 1401 e o mais robusto 7094 estava tornando extremamente custoso para os fabricantes manter a produção deles. Portanto, foi necessária a criação de um novo sistema operacional que fosse possível que rodasse os mesmos softwares em qualquer máquina que houvesse aquele Sistema Operacional. Então foi criado o OS/360 que era uma série de máquinas cujos softwares eram compatíveis. A diferencia das máquinas estavam principalmente nos preços e no desempenho (quantidade máxima de memória, velocidade do processador, quantidade de dispositivos de E/S e etc.)

Com a adesão dos CIs, mesmo que em pequenas escalas, os computadores passaram a ter um melhor custo-benefício em comparação ao da geração passada com apenas transistores individuais.

Como os requisitos de funcionamento de um único software era extremamente alto, pois ele deveria ser melhor que na 1401 na leitura de entrada e na saída, melhor que o 7094 na execução do programa, tinha que ser eficiência independente de quantos periféricos o sistema, foi criado um Sistema Operacional extremamente complexo, com milhões de linhas escritas na linguagem assembly por milhares de programadores. Com isso, o Sistema operacional havia milhares de erros e necessitava da continuação da linha de computadores para novas versões para que pudessem ser tratados, porém ao tratar um erro, novos surgiam.

Várias técnicas foram criadas para correção de diferentes erros, e uma das mais importantes foi a multiprogramação. Antigamente, a execução de algumas tarefas pelo 7094 era interrompida, permanecendo ociosa, para aguardar o término da leitura de uma fita no dispositivo de E/S ou então até que a E/S terminasse. E isso era responsável pela maior parte do tempo de execução das tarefas, essa espera. Para resolver esse problema, foi pensado em uma maneta de múltiplas tarefas serem executadas ao mesmo tempo. Com isso, foi pensado em separar a memória em diversas partes para que alguma tarefa ociosa pudesse ser armazenada,

liberando assim o CPU para executar uma outra tarefa independente. Dessa forma, o CPU permaneceria praticamente 100% em execução, não ficando mais parado enquanto a tarefa estivesse ociosa. Isso reduziu drasticamente o tempo de execução de um conjunto de tarefas.

Uma outra variante que foi tratada foi o *leaves and takes* de fitas magnéticas que eram feita antigamente, fazendo assim não necessário o uso do 1401 mais, pois com a técnica denominada **spooling**, quando uma tarefa era finalizada, o sistema operacional era capaz de carregar, a partir de novos cartões perfurados, novas tarefas e gravá-las em fitas magnéticas, utilizando o espaço em disco que não mais está sendo utilizado, portanto foi liberado, daquela tarefa que, anteriormente, havia sido finalizada.

Outro problema foi o tempo que levava para submeter uma tarefa e obter a saída poderia levar horas, e uma simples vírgula poderia causar erro de compilação que levaria o dia todo para ser corrigido. Então foi criado o **time-sharing** que basicamente permitia que mais de 1 usuário conectasse por meio de um terminal online. Esses usuários então teriam acesso à CPU por um tempo determinado e esse acesso seria ciclicamente passado para qualquer um, para execução de instruções curtas de correção como depurar o código. A medida que alguns usuários conectados fiquem ociosos, o tempo é ciclicamente alocado a cada um dos que estão requisitando o uso do CPU. Além disso, quando a CPU estaria ociosa, ela executava programas maiores em background.

- 4ª Geração: A quarta geração abrange o período de 1980 até os dias de hoje e foi marcada, principalmente, pelo desenvolvimento de circuitos integrados em larga escala, contando milhares de transistores em um centímetro quadrado. É nessa geração, também, que surgiu a era dos computadores pessoais. Essa geração abrange toda a era do surgimento da Microsoft, da Apple e dos sistemas operacionais mais utilizados no mundo, como o UNIX e o Windows XP e o Vista. Bem como, obviamente, os Sistemas Operacionais atuais e os computadores como conhecemos hoje.

Q.2 - Você deverá me explicar, vídeo de, no máximo, 3 minutos, como uma interrupção de

hardware ocorre no 8085 ligado no RST 7 , em nível dos registradores PC, SP, etc.. até um a instrução RET . Explique uma pessoa apertando um tecla, que pressionada, vai acionar a interrupção RST 7.

O 8085 contém 4 instruções de recomeço por hardware (interrupções externas). Quando são ativadas, os circuitos internos produzirão uma chamada de subrotina por hardware que desvia o processamento para um endereço predeterminado.

As operações de entrada e saída por interrupção usam uma das interrupções por hardware de forma que, quando o periférico está pronto para a transferência de dados a interrupção selecionada é acionada, permitindo a transferência de dados de forma mais rápida.

As interrupções externas têm ordem de prioridade, no caso de dois ou mais pedidos de interrupções

Ordem de prioridade:

Interrupção	Prioridade	Posição do Vetor
TRAP	1	0024 h
RST 7.5	2	003C h
RST 6.5	3	0034 h
RST 5.5	4	002C h
INTR	5	Nenhuma

o RST 7.7 pode ser "mascarado" ou seja, pode ser desativado por software

- Para evitar que o mP seja interrompido quando está a executar código crítico
- Para que uma rotina de interrupção não se interrompa a si própria
- Para ignorar temporariamente os pedidos de um dado periférico