



ARQUITETURA DE COMPUTADORES

AULA 5



Prof. André Roberto Guerra



CONVERSA INICIAL

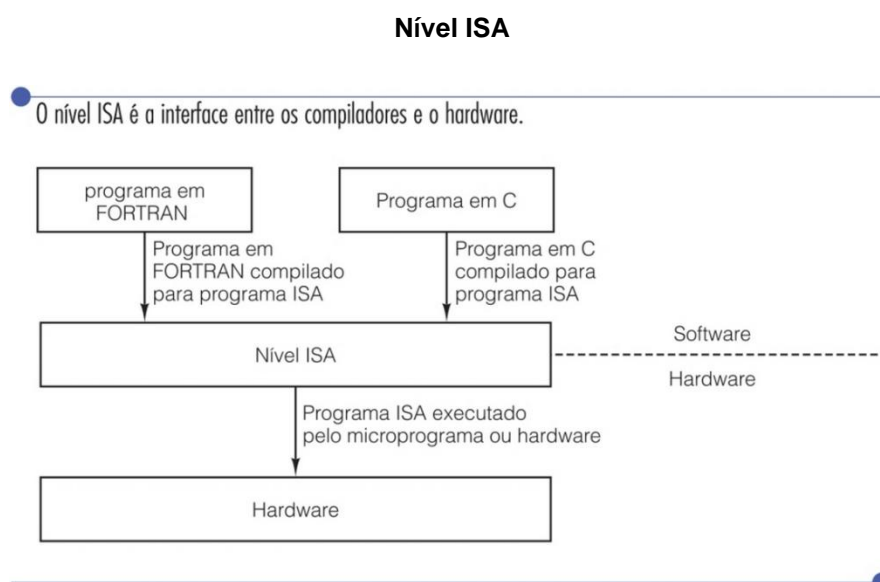
Olá, seja bem-vindo a quinta aula de Arquitetura de Computadores!

Nessa aula veremos os conteúdos que descrevem os sistemas operacionais e a Arquitetura do Conjunto de Instrução - ISA:

- Visão geral ISA
- Tipos de dados ISA
- Formatos, endereçamento e tipos de instruções ISA
- Visão geral de sistema operacional
- Objetivos e funções de sistemas operacionais
- Tipos e exemplos de sistemas operacionais

TEMA 1 - VISÃO GERAL DO ISA

A arquitetura do conjunto de instrução (ISA - *Instruction Set Architecture*) está entre o nível da microarquitetura e o nível do sistema operacional. Apresenta-se ao mesmo tempo como fronteira e/ou interface entre o *hardware* e o *software*, significado especial que o torna importante para arquitetos de sistema. Veja na imagem a seguir:



(TANENBAUM, Andrew S., **Organização Estruturada de Computadores**. Editora Phb, 6ª Ed.)



Historicamente, o ISA foi desenvolvido antes de quaisquer outros níveis e, originalmente era o único nível. Até hoje é denominado como “a arquitetura” da máquina ou (incorretamente) como “linguagem de montagem”.

Embora o *hardware* possa executar diretamente programas escritos em linguagem de alto nível (C, C++, Java), para tornar-se útil deve ser heterogêneo e executar programas escritos nas mais diversas linguagens, e também fazer uso da vantagem em desempenho da compilação em relação à interpretação.

O nível ISA é uma abordagem adotada por projetistas de sistemas para traduzir programas escritos nas mais diversas linguagens de alto nível para uma forma intermediária comum, e para construir *hardware* que os execute diretamente nesta forma. O nível ISA define a interface entre os compiladores e o *hardware*.
É a linguagem que ambos têm de entender.

Compatibilidade

O ideal é que, ao projetar uma nova máquina, os arquitetos, os escritores de compiladores e os engenheiros de *hardware* debatam juntos sobre as características do nível ISA necessárias a cada um. Contudo, na realidade, quando surge uma nova máquina, alguns questionamentos são feitos pelos clientes potenciais:

- Ela é compatível com sua antecessora?
- Ela pode executar meu sistema operacional antigo?
- Ela executará todos os meus programas de aplicação existentes sem modificação?

Poucos se importam com o *hardware* ou sabem o que ele faz, querem apenas que seja garantida a compatibilidade com seus aplicativos e sistemas. O desafio é projetar máquinas melhores e mais velozes garantindo a compatibilidade com sistemas e aplicativos antigos.

Dois fatores são essenciais para uma “boa” ISA:

1. Definir um conjunto de instruções que possa ser facilmente implementado, com tecnologias atuais e futuras, contentando aos projetistas de *hardware*;



2. Fornecer um alvo claro para o código compilado, contentando os projetistas de *software* na geração de códigos para ela

O ISA é definido pelo conjunto formado pelo modelo de memória utilizado pela máquina, a quantidade e o tipo dos registradores, os tipos de dados e instruções disponíveis, bem como o modo como a máquina apresenta-se ao programador e o código ISA que o compilador produz.

TEMA 2 - TIPO DE DADOS ISA

Todos os computadores precisam de dados. Muitos sistemas de computação têm como propósito único o processamento de dados (financeiros, comerciais, científicos, de engenharia ou outros) e eles (dados) têm de ser representados de forma específica no interior do computador. A questão fundamental é se há (ou não) suporte de *hardware* para um tipo particular de dados.

No nível ISA, são utilizados os tipos de dados **numéricos** e **não numéricos**. A maioria dos dados numéricos são os inteiros (com e sem sinal), inteiro decimal em código binário e ponto flutuante. E os não numéricos são os ponteiros (endereço de máquina), caracteres ASCII e UNICODE e os dados booleanos (de dois valores: V ou F, 0 ou 1, M ou F, etc.).

Um suporte de *hardware* significa que uma ou mais instruções esperam dados em um formato particular e o usuário não tem liberdade de escolher um formato diferente.

TEMA 3 - FORMATOS, ENDEREÇAMENTO E TIPOS DE INSTRUÇÃO ISA

Instruções ISA são formadas por *opcodes* (códigos de operação), usualmente em conjunto com alguma informação adicional, tais como de onde vêm os operandos e para onde vão os resultados.

São diversos os formatos possíveis de instrução, contudo instruções sempre têm um *opcode* que indica o que ela faz. Instruções podem não ter endereço ou ter



um, dois e ou três. Em algumas máquinas todas instruções têm o mesmo comprimento; já em outras pode haver muito comprimentos diferentes.

Vários fatores devem ser considerados na escolha do formato de instruções e a dificuldade dessa decisão não deve ser subestimada e deve ser tomada no início do projeto. Se ele for um sucesso comercial, o conjunto de instruções pode sobreviver por muitos anos. A capacidade de acrescentar novas instruções e de explorar outras oportunidades que surgem é de grande importância e depende (muito) da tecnologia de implementação da máquina.

A maioria das instruções possuem operandos e o **endereçamento** é o tópico geral que especifica onde eles (operandos) estão. Os modos de endereçamento são o modo como os *bits* de um campo de endereço são interpretados para encontrar o operando.

São diversos os modos, clique nos botões e descubra!

1. Endereçamento imediato (o modo mais simples)
2. Endereçamento direto (especifica o endereço completo na memória)
3. Endereçamento de registrador (especifica um registrador em vez de uma localização na memória)
4. Endereçamento indireto de registrador ou ponteiro (endereço contido em um registrador. Operando vem ou vai para memória, sem endereço na instrução)
5. Endereçamento indexado (fornece um registrador (explícito ou implícito) e um deslocamento constante)
6. Endereçamento de base indexado (dois registradores: base e índice)
7. Endereçamento de pilha (infixa, pós-fixa ou notação polonesa invertida)

Instruções de nível ISA podem ser divididas em aproximadamente meia dúzia de grupos que guardam relativa similaridade de uma máquina para outra, ainda que diferentes nos detalhes. Todo computador tem algumas instruções fora do comum, acrescentadas para manter a compatibilidade com modelos anteriores, ou porque o arquiteto teve uma ideia brilhante, ou possivelmente porque algum fabricante foi pago para inseri-la.

Dentre elas destacam-se:



- Instruções para movimento de dados
- Operações diádicas
- Operações monádicas
- Comparações e desvios condicionais
- Instruções de chamada de procedimento
- Controle de laço
- Entrada/Saída

TEMA 4 - VISÃO GERAL DE SISTEMA OPERACIONAL

Segundo Tanenbaum:

Sistema Operacional é um programa que, do ponto de vista do programador, acrescenta uma variedade de novas instruções e características, acima e além do que o nível ISA fornece. Normalmente, o sistema operacional é implementado, em grande parte, em *software*, mas não há nenhuma razão teórica por que ele não possa ser colocado em *hardware*, exatamente como acontece com os microprogramas (quando estão presentes).

Para saber mais sobre essa definição e outros conceitos, acesse o Único e consulte o livro “Arquitetura e Organização de Computadores”.

Já Stallings define o **Sistema Operacional como um programa que gerencia os recursos do computador, fornece serviços para os programadores e estabelece uma ordem de execução de outros programas.**

Para nós, é essencial certo conhecimento sobre sistemas operacionais, para que possamos entender os mecanismos pelos quais a CPU controla o computador. Em particular, o efeito das **interrupções** e o **gerenciamento da hierarquia de memória**.

O Sistema Operacional (SO) é o *software* que controla a execução de programas em um processador e **gerencia os recursos do computador**. Diversas funções desempenhadas pelo SO, incluindo as duas funções mais relevantes para o estudo da arquitetura e organização de computadores: o **escalonamento de processos** e o **gerenciamento de memória**, que só podem ser executadas de modo rápido e eficiente se o SO dispuser de um suporte adequado do *hardware* do processador.



Quase todos os processadores dispõem desse suporte, em maior ou menor extensão, incluindo *hardware* de **gerenciamento de memória virtual** e de **gerenciamento de processos**. Isso inclui registradores de propósito especial e áreas de armazenamento temporário, além de um conjunto de circuitos para realizar tarefas básicas de gerenciamento de recursos.

O **escalonamento de processos** ou tarefas é uma das funções mais importantes do SO. O SO determina quais processos devem ser executados a cada instante. Tipicamente, o *hardware* interrompe periodicamente o processo que está sendo executado para permitir ao SO realizar uma nova decisão de escalonamento. Isso possibilita compartilhar o tempo do processador entre um determinado número de processos de modo imparcial.

O **gerenciamento de memória** é outra importante função do sistema operacional.

A maioria dos sistemas operacionais atuais inclui a capacidade de **memória virtual**, o que traz dois benefícios:

1. Um processo pode ser executado na memória principal sem que todas as instruções e dados do programa estejam armazenados na memória principal; e
2. O espaço de memória total disponível para um programa pode exceder o tamanho da memória principal do sistema.

Embora o gerenciamento de memória seja feito por *software*, o sistema operacional conta com suporte do *hardware* do processador, incluindo *hardware* de paginação e de segmentação da memória.

TEMA 5 - OBJETIVOS E FUNÇÕES DE SISTEMAS OPERACIONAIS

Stallings descreve em sua obra que o sistema operacional é um programa que controla a execução de programas aplicativos e age como **interface entre o usuário e o hardware do computador**.

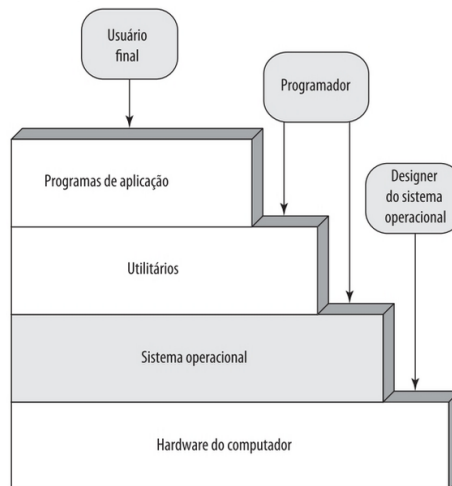
Ele possui basicamente dois objetivos:

- **Conveniência:** um sistema operacional visa tornar o uso do computador mais conveniente.



- **Eficiência:** um sistema operacional permite uma utilização mais eficiente dos recursos do sistema.

O *hardware* e o *software* usados para fornecer aplicações aos usuários podem ser vistos sob a forma de uma organização em camadas ou hierárquica, como representado abaixo:



O usuário dessas aplicações, o usuário final, geralmente não está interessado na arquitetura do computador. Dessa maneira, ele vê o sistema de computação em termos de uma aplicação.

Essa aplicação é escrita em uma linguagem de programação e desenvolvida por um programador de aplicações. Se esses programas aplicativos tivessem de ser escritos usando o conjunto de instruções do processador e, além disso, tivessem também de controlar o *hardware* do computador, a tarefa de desenvolver programas seria extremamente complexa.

Para facilitar essa tarefa, existe um conjunto de programas de sistema, alguns desses conhecidos como utilitários. Eles implementam funções usadas frequentemente que auxiliam na criação de programas, gerenciamento de arquivos e controle de dispositivos de E/S (Entradas e Saídas). Um programador usa esses recursos para desenvolver uma aplicação, que, ao ser executada, invoca os utilitários para desempenhar certas funções.

Você sabia que o programa de sistema mais importante é o sistema operacional? Ele esconde os detalhes do *hardware* do programador, fornecendo uma interface conveniente para o uso do sistema. Ele age como um mediador, tornando mais



fácil para o programador e para os programas aplicativos acessar e usar esses recursos e serviços.

TROCANDO IDEIAS

Chegou a hora de trocar conhecimentos, vamos lá!

Comente a sua experiência com os diversos tipos de sistemas operacionais, fale um pouco também sobre a compatibilidade dos sistemas.

NA PRÁTICA

O ISA é apresentada como a interface entre *hardware* e *software*; já os sistemas operacionais apresentados como a interface entre o usuário e a máquina.

Pesquise através dos diversos canais sobre os novos e atuais microprocessadores e sistemas operacionais, enfatizando a compatibilidade entre eles e com os sistemas e aplicações antigos, validando os conceitos estudados.

Desse modo, você poderá ver como as informações que vimos nessa aula, podem ser aplicadas na prática!

SÍNTESE

Na aula de hoje vimos que o nível de arquitetura do conjunto de instrução é o que a maioria das pessoas chama de **linguagem de máquina**.

Aprendemos que o Sistema Operacional pode ser considerado como intérprete para certas características de arquitetura não encontradas no nível ISA e que entre as principais estão memória virtual, instruções de E/S virtual e facilidades de processamento paralelo.

COMPARTILHANDO

Gostou do conteúdo da nossa aula? Então compartilhe com seus colegas e familiares os conceitos e curiosidades vistos.

Não esqueça de mencionar o que você achou de mais importante e como pode aplicar esses conhecimentos no dia a dia!