

Aula 00

*Arquitetura e Sistemas Operacionais p/
Receita Federal (Analista de Informática)
2020.2 - Pré-Edital*

Autor:
**Equipe Informática e TI, Evandro
Dalla Vecchia Pereira**

27 de Julho de 2020

Sumário

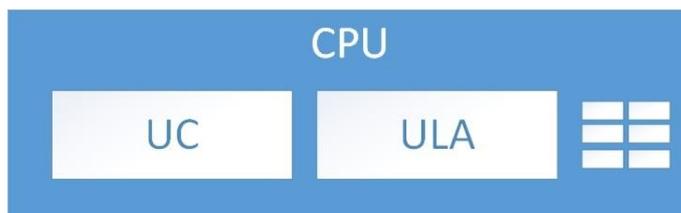
Processador (CPU)	2
Questões Comentadas	3
Arquiteturas de Processadores (RISC e CISC)	7
Conceitos	7
Questões Comentadas	9
Linguagens de Máquina e de Montagem	13
Conceitos	13
Instrução de Máquina.....	14
Representação da Instrução.....	14
Modos de Endereçamento.....	15
Linguagem de Montagem	17
Questões Comentadas	20
Lista de Questões.....	23
Gabarito.....	30



Processador (CPU)

O **processador** (CPU - *Central Processing Unit*) é o “cérebro” do computador, tendo como função a execução de programas armazenados na memória principal. Basicamente a CPU busca as instruções, examina-as e as executa! Os componentes básicos da CPU são:

- Unidade de Controle (UC);
- Unidade Lógica e Aritmética (ULA);
- Registradores;
- Interconexão da CPU: barramentos que proporcionam a comunicação entre a UC, ULA e os registradores.



A **Unidade de Controle (UC)** é responsável por receber instruções pelo barramento de instruções. As instruções vêm da memória de acordo com o endereço enviado pela UC para a memória através do barramento de endereço das instruções. A Unidade de Controle não executa as instruções. Ela as lê, decodifica e passa os comandos para a UCD (Unidade de Ciclo de Dados, não veremos essa unidade, pois não cai em prova de concurso) determinando como as instruções devem ser executadas e com quais dados.

Baseada nesses comandos, a UCD pode buscar os dados necessários na memória, executar as devidas operações e enviar o resultado de volta para a memória para ser armazenado. Tudo é controlado de acordo com os comandos internos enviados pela UC, que por sua vez se baseia na instrução decodificada.

Tudo isso é controlado por um sinal síncrono de relógio (*clock*). A cada *clock* a unidade sabe que deve executar um passo, passar os dados para quem deve, e se preparar para o próximo passo. Obviamente que quanto mais rápido for o relógio, mais operações por segundo o processador consegue executar. A velocidade do relógio é medida em frequência, utilizando a unidade Hertz (Hz).

Um Hertz significa um passo por segundo. **Os processadores atuais trabalham na faixa dos poucos GHz** (bilhão de Hertz, ou de passos por segundo). As instruções podem ser simples (adição, subtração etc.), mas fazem isso em uma grande velocidade!

A **Unidade Lógica e Aritmética (ULA)** efetua operações, como adição, subtração, “E” lógico (AND) etc., sobre suas entradas, produzindo um resultado no registrador de saída. Depois, esse valor pode ser escrito na memória principal, se for desejado.



Grande parte das instruções pode ser classificada como registrador-memória ou registrador-registrador. Instruções registrador-memória permitem que palavras¹ de memória sejam buscadas diretamente para a ULA e armazene da ULA diretamente em memória. Instruções registrador-registrador buscam dois operandos nos registradores para a ULA, efetua a operação e armazena o resultado em um dos registradores.

Os **registradores** são pequenas memórias de alta velocidade que ficam dentro da CPU. Eles armazenam resultados temporários e o controle de informações, sendo que alguns são de uso geral e outros de uso específico. O registrador mais importante é o PC (*Program Counter*), que indica a próxima instrução a ser buscada para execução. Outro importante é o IR (*Instruction Register*), que mantém a instrução que está sendo executada;

Vamos ver o passo a passo do ciclo conhecido como **buscar-decodificar-executar**:

1. Traz a próxima instrução da memória para o registrador de instrução (IR);
2. Altera o registrador contador de programa (PC) para que aponte para a próxima instrução;
3. Determina o tipo de instrução trazida;
4. Se a instrução utilizar uma palavra na memória, determinar onde ela está;
5. Traz a palavra para um registrador da CPU, se necessário;
6. Executa a instrução;
7. Volta à etapa 1, para a execução da próxima instrução.

Questões Comentadas

1. (CESPE/Polícia Federal - 2004) Para se determinar a capacidade de processamento e saber qual é o computador de melhor desempenho, é suficiente consultar a frequência do relógio (clock) do processador.

Comentários:

Não adianta ter um clock de 3 GHz e apenas um processador, por exemplo. Se tiver 2 ou mais processadores e um clock menor (2,5 GHz), o desempenho certamente será melhor (se o restante dos componentes for equivalente: memória, barramento, entre outros). Portanto, a questão está **errada**.

2. (MS CONCURSOS/CODENI-RJ - 2010) É o componente vital do sistema, porque, além de efetivamente realizar as ações finais, interpreta o tipo e o modo de execução de uma instrução, bem como controla quando e o que deve ser realizado pelos demais componentes, emitindo para isso sinais apropriados de controle. A descrição acima refere-se a?

- A) Dispositivos de Entrada e Saída.
- B) Memória Principal.

¹ Palavras são as unidades de dados movimentadas entre a memória e os registradores. Ex.: palavra de 64 bits.



- C) Memória Secundária.
- D) Unidade Central de Processamento.

Comentários:

“Quem” realiza o processamento dos dados, bem como o devido controle dos dados a serem carregados em memória, buscados para o processador, entre outras atividades, é o processador (também conhecido por CPU – Unidade Central de Processamento). Portanto, a **alternativa D está correta e é o gabarito da questão.**

3. (AOCP/TCE-PA - 2012) Em computação CPU significa

- A) Central de Processamento Única.
- B) Único Centro de Processamento.
- C) Unidade Central de Processamento.
- D) Central da Unidade de Processamento.
- E) Centro da Unidade de Processamento.

Comentários:

CPU = Central Processing Unit (Unidade Central de Processamento). Portanto, a **alternativa C está correta e é o gabarito da questão.**

4. (IF-PA/IF-PA - 2016) A Unidade Central de Processamento (UCP) é composta por um conjunto de componentes básicos, EXCETO:

- A) Unidade de Controle.
- B) Unidade de Entrada/Saída.
- C) Unidade de Aritmética e Lógica.
- D) Conjunto de Registradores.
- E) Chipset.

Comentários:

O processador (CPU): é o “cérebro” do computador, tendo como função a execução de programas armazenados na memória principal. Basicamente a CPU busca as instruções, examina-as e as executa! Os componentes básicos da CPU são:

- Unidade de Controle (UC): busca instruções na memória principal e determina seu tipo;



- Unidade Lógica e Aritmética (ULA): efetua operações, ex.: adição, “E” lógico (AND), etc.;
- Registradores: pequenas memórias de alta velocidade que ficam dentro da CPU. Armazena resultados temporários e controle de informações, sendo que alguns são de uso geral e outros de uso específico;
- Interconexão da CPU: mecanismos que proporcionam a comunicação entre a UC, ULA e os registradores.

Portanto, a **alternativa E está correta e é o gabarito da questão.**

5. (CESPE/TCE-PA - 2016) Uma das funções de uma unidade central de processamento é buscar instruções de programas armazenados na memória principal, examiná-las e executá-las uma após a outra.

Comentários:

Mais uma vez... “O processador (CPU): é o “cérebro” do computador, tendo como função a execução de programas armazenados na memória principal. Basicamente a CPU busca as instruções, examina-as e as executa!”. Portanto, a questão está **correta**.

6. (INAZ do Pará/DPE-PR - 2017) O funcionário de uma empresa precisa adquirir um novo computador. Durante suas pesquisas, ele se interessou por um computador com a seguinte configuração dos componentes de hardware: 3,5 GHz, 4 GB, 1 TB, 64 bits. Nessa configuração,

- A) 64 bits é a taxa de transmissão da porta USB.
- B) 4 GB é a quantidade da memória ROM.
- C) 1 TB é a capacidade de memória RAM.
- D) 3,5 GHz é a velocidade do processador.

Comentários:

3,5 GHz indica a velocidade do processador (3,5 bilhões de passos por segundo). 4 GB indica a capacidade da memória RAM. 1 TB indica a capacidade de armazenamento do HD. 64 bits indica o tamanho dos registros do processador. O registro de um processador é o local onde ele armazena os "endereços" dos dados que ele precisa acessar mais rapidamente para funcionar bem. Se o seu processador for 64 bits, é melhor instalar um sistema operacional de 64 bits também, para que ele possa funcionar com o máximo de sua capacidade. Processadores de 64 bits podem rodar sistemas operacionais de 32 bits, mas só poderão acessar 4GB de RAM (232), e terão um desempenho inferior. Portanto, a **alternativa D está correta e é o gabarito da questão.**

7. (FGV/AL-RO - 2018) Assinale a opção que indica os componentes de uma unidade central de processamento ou CPU (Central Processing Unit).

- A) Unidade lógica e aritmética, unidade de controle e registradores.
- B) Discos ópticos, disco rígido e drive.



C) Scanner, plotter e dispositivos de entrada.

D) Memória ROM, memória RAM e cache.

E) Mouse, teclado e impressora.

Comentários:

Unidade Lógico-Aritmética (ULA): é responsável pelos cálculos lógicos e matemáticos do computador. Sempre ouvimos dizer que é a CPU quem faz os cálculos, porém, se olharmos com detalhamento, veremos que é a ULA o dispositivo da CPU que executa todas essas operações. A ULA é a responsável por todas as operações aritméticas e lógicas, tais como soma, subtração, multiplicação, divisão, e as operações lógicas, como AND, OR, NOT, XOR, etc.

Registradores: a ULA executa cálculos rápidos demais para enviarem para a RAM, e os resultados desses cálculos precisam ser armazenados em algum lugar para uso da própria ULA. O local de armazenamento temporário desses resultados são os registradores. É uma memória interna do núcleo de processamento que trabalha exclusivamente para a ULA. Podemos afirmar que a memória mais rápida do computador, que possui a tecnologia SRAM (memória RAM volátil estática).

Unidade de Controle: é o dispositivo interno da CPU responsável pelo controle dos fluxos de dados entre ULA e Registradores e vice-versa, além de controlar os demais dados que circulam dentro do processador.

Decodificadores: sempre é necessário quebrar instruções complexas em instruções mais simples para processar dados. Essa é a função dos decodificadores da CPU.

Portanto, a **alternativa A está correta e é o gabarito da questão.**

8. (CESPE/IFF - 2018) A respeito da unidade central de processamento (CPU), julgue os itens que se seguem.

I A CPU, também denominada processador, tem como função controlar a operação do computador.

II Os registradores são responsáveis por oferecer armazenamento interno à CPU.

III A unidade de controle e a unidade aritmética e lógica fazem parte da CPU.

Assinale a opção correta

A) Apenas o item I está certo.

B) Apenas o item II está certo.

C) Apenas os itens I e III estão certos.

D) Apenas os itens II e III estão certos.

E) Todos os itens estão certos.



Comentários:

(I) A CPU é o “cérebro”, quem processa as instruções e dados! (II) Os registradores podem ser chamadas de memórias internas do processador, neles ficam, por exemplo, os dados a serem calculados; (III) O processador possui unidade de controle (UC), unidade lógica e aritmética (ULA), além de registradores e uma interconexão desses 3 elementos. Portanto, todas estão corretas (**alternativa E**).

Arquiteturas de Processadores (RISC e CISC)

Conceitos

Quando o assunto é saber qual a melhor arquitetura de processador, sempre há polêmica. Muitos defendem que os “Macs” são mais rápidos por terem chips RISC, por exemplo. Mas o que é RISC? E CISC? Quais vantagens e desvantagens? Vamos lá...

Um processador **CISC** (*Complex Instruction Set Computer* - Computador com um Conjunto Complexo de Instruções), é capaz de executar **várias centenas de instruções complexas diferentes**, sendo extremamente versátil. Alguns exemplos de processadores CISC são o 386 e o 486.

Alguns fabricantes decidiram seguir o caminho contrário, criando o padrão **RISC** (*Reduced Instruction Set Computer* - Computador com um Conjunto Reduzido de Instruções). Os processadores RISC são capazes de executar apenas **algumas poucas instruções simples**. Justamente por isso, os chips baseados nesta arquitetura são **mais simples e muito mais baratos**.

Outra vantagem dos processadores RISC, é que, por terem um menor número de circuitos internos, podem trabalhar a frequências mais altas. Alguns exemplos de processadores CISC são Sparc (Sun), Mips (Silicon Graphics), Power (IBM) e Alpha (DEC).

Aí surge a dúvida...como um chip que é capaz de executar algumas poucas instruções pode ser considerado por muitos, mais rápido do que outro que executa centenas delas? A grande questão é que um processador RISC é capaz de executar suas poucas instruções muito mais rapidamente. A ideia principal é que apesar de um processador CISC ser capaz de executar centenas de instruções diferentes, apenas algumas são usadas frequentemente, o que parece ser um desperdício, não?

Mas, uma coisa é garantida: em instruções complexas os processadores CISC se saem melhor! O que podemos concluir, então? O ideal é fazer um mix das duas tecnologias, e é por isso que atualmente temos processadores híbridos, que são essencialmente processadores CISC, mas incorporam muitas características dos processadores RISC (ou vice-versa). Tanto os processadores da família x86, como o Pentium II, Pentium III e AMD Athlon, quanto processadores supostamente RISC, como o MIPS R10000 e o HP PA-8000 misturam características das duas arquiteturas, por simples questão de desempenho.

Uma coisa é o mundo real, outra é o mundo dos concursos, onde é importante saber diferenciar bem as características RISC e CISC.

Vamos começar pela **CISC** (*Complex Instruction Set Computer*):



- Possui grande quantidade de instruções, com múltiplos modos de endereçamento;
- O conceito de microprogramação facilitou o projeto de instruções complexas;
- Microcódigo reside em memória de controle (memória ROM que fica dentro da Unidade de Controle do processador, bem mais rápido que a memória RAM!);
- Criação de novas instruções quase não tem custo (basta ter espaço ainda na memória de controle).

Essas características facilitam a implementação do conceito de famílias de processadores. Por exemplo, na arquitetura x86 houve um acréscimo de instruções do 386 para 486, Pentium, Pentium MMX etc. Ou seja, instruções novas foram introduzidas, aproveitando as antigas, sem ter que começar do zero, sem alterar o projeto básico!

Podemos destacar três aspectos básicos:

- Uso de microcódigo (camada de hardware em nível de instruções ou estruturas de dados envolvidas na implementação do nível superior de código de máquina);
- As instruções são completas e eficientes;
- Instruções de máquina de “alto nível” (complexidade semelhante à dos comandos de alto nível).

Algumas características:

- Formato de 2 operandos é o mais comum, ex.: ADD AX, mem;
- Uso dos modos: Registrador para registrador, Registrador para memória, Memória para registrador;
- Múltiplos modos de endereçamento para a memória, incluindo indexação (vetores);
- Instruções com largura variável;
- Instruções requerem múltiplos ciclos de relógio para completar a execução, ex.: se existe a busca de dois operandos na memória, demora mais;
- Poucos registradores, devido ao pouco espaço no chip (tem memória para o microcódigo, decodificador etc.) e à possibilidade de acesso a operandos na memória;
- Há registradores especializados: controle (*flags*), segmento (ponteiro da pilha) etc.

Agora vamos para a arquitetura **RISC (*Reduced Instruction Set Computer*)**:

- Possui poucas instruções e todas possuem a mesma largura;
- Execução otimizada de chamada de funções;
- Menor quantidade de modos de endereçamento;
- Uso intenso de *pipelining*, pois é mais fácil implementar o paralelismo quando se tem instruções de mesmo tamanho;
- Execução rápida de cada instrução (uma por ciclo de relógio);
- Processadores RISC não requerem microcódigos (sobra mais espaço no chip);
- Menos acesso à memória principal, instruções que acessam a memória: LOAD e STORE (arquitetura registrador – registrador, ou seja, após buscar os dados da memória e colocá-los em registradores, as operações são realizadas);
- Maior quantidade de registradores, justamente pelo explicado no item anterior.

Vamos analisar a tabela abaixo, de diferentes processadores e na sequência vamos classifica-los como RISC ou CISC.



Características	MIPS R4000	RS/6000	VAX11/780	INTEL 486
Quantidade de instruções	94	183	303	235
Modos de endereçamento	1	4	22	11
Largura de instruções (bytes)	4	4	2-57	1-12
Quantidade de registradores de uso geral	32	32	16	8

Olhando pela quantidade de instruções, o que apresenta 94 parece ser RISC (poucas instruções) e o 303 CISC (muitas instruções), mas os outros dois são próximos e fica a dúvida.

Analisando os modos de endereçamento, fica evidente que os dois primeiros são RISC (poucos modos), enquanto os dois últimos possuem bem mais.

Pela largura de instruções fica mais claro ainda que os dois primeiros são RISC, pois possuem uma largura fixa de instruções (4 bytes), enquanto os outros dois possuem instruções de diversos tamanhos (2 a 57 bytes um deles e o outro entre 1 e 12 bytes).

E para arrematar nossa análise, os dois que achamos que são RISC possuem mais registradores (32 cada um deles), enquanto os outros dois processadores possuem menos registradores (um possui 16 e o outro 8).

Conclusão: os processadores MIPS R4000 e RS/6000 possuem arquitetura RISC e os processadores VAX11/780 e INTEL 486 possuem arquitetura CISC. Lembrando...essa é uma classificação conforme as características que predominam, mas na realidade os processadores não possuem características apenas de um tipo de arquitetura e são chamados de híbridos.

Questões Comentadas

9. (ESAF/SUSEP - 2010) Em uma Arquitetura RISC

- A) há poucos registradores.
- B) há pouco uso da técnica pipelining.
- C) as instruções possuem diversos formatos.
- D) as instruções são realizadas por microcódigo.
- E) as instruções utilizam poucos ciclos de máquina.

Comentários:



Em uma arquitetura RISC existem muitos registradores, há muito uso da técnica pipelining (devido ao tamanho fixo das instruções), as instruções possuem poucos formatos, não são realizadas por microcódigo. E por fim, as instruções utilizam poucos ciclos de máquina (um ciclo, na verdade)! Portanto, a **alternativa E está correta e é o gabarito da questão.**

10.(FCC/TRE-AM - 2010) Numa máquina estruturada multinível, é o nível essencial para as máquinas CISC (Complex Instruction Set Computer), mas que inexistente nas máquinas RISC (Reduced Instruction Set Computer). Trata-se do nível

- A) do sistema operacional.
- B) de lógica digital.
- C) de microprogramação.
- D) convencional de máquina.
- E) do montador.

Comentários:

A microprogramação é utilizada pela arquitetura CISC, o que consome espaço no chip (na unidade de controle do processador), algo que não existe na arquitetura RISC. Portanto, a **alternativa C está correta e é o gabarito da questão.**

11.(CESPE/Correios - 2011) As instruções CISC são mais simples que as instruções RISC, por isso, os compiladores para máquinas CISC são mais complexos, visto que precisam compensar a simplificação presente nas instruções. Entretanto, se for usado pipeline, a complexidade do compilador CISC é reduzida, pois a arquitetura pipeline evita a necessidade de reordenação inteligente de instruções.

Comentários:

O nome já deixa claro: “Complex Instruction Set Computer”, portanto são mais complexas. Os compiladores para máquinas RISC é que são mais complexos, pois devem lidar com instruções simples. Portanto, a questão está **errada.**

12.(VUNESP/UNESP - 2013) Um computador baseado em uma Unidade Central de Processamento do tipo RISC

- A) não faz uso de pipeline.
- B) executa cada instrução em um ciclo de relógio
- C) possui instruções de tamanho variável.
- D) possui muitos modos de endereçamento



E) possui um grande conjunto de instruções.

Comentários:

Processadores RISC possuem instruções de tamanho fixo e cada instrução é executada em um ciclo de relógio. Portanto, a **alternativa B está correta e é o gabarito da questão**.

13.(FUNDEP/IPSEMG - 2013) A arquitetura RISC de um computador possui as seguintes características, EXCETO:

- A) Formatos simples de instruções.
- B) Modos simples de endereçamento.
- C) Operações memória-para-memória.
- D) Uma instrução por ciclo.

Comentários:

RISC é tudo “simples” e uma instrução por ciclo de relógio. Realiza operações registrador-registrador, ou seja, tem que buscar da memória os dados antes (através de LOAD). Portanto, a **alternativa C está correta e é o gabarito da questão**.

14.(CESPE/Antaq - 2014) Atualmente, os fabricantes de computadores têm adotado exclusivamente a arquitetura RISC para o desenvolvimento de chips para processadores, dado o melhor desempenho dessa arquitetura em relação à arquitetura CISC.

Comentários:

Esse “exclusivamente” mata, heim! Para começar os fabricantes têm utilizado uma arquitetura híbrida, com mais características de uma ou de outra. Portanto, a questão está **errada**.

15.(IADES/PCDF - 2016) Em relação ao projeto de máquinas RISC e CISC, assinale a alternativa correta.

- A) Dadas as características das instruções das máquinas CISC, o pipeline fica favorecido nessa arquitetura.
- B) Arquiteturas RISC normalmente realizam poucas operações de registrador para registrador, aumentando o acesso à memória cache.
- C) Programas para arquiteturas CISC sempre possuem tamanho menor que programas para arquiteturas RISC, devido à relação um para um de instruções de máquina e instruções de compilador.
- D) Arquiteturas RISC tendem a enfatizar referências aos registradores no lugar de referências à memória.



E) Arquiteturas CISC usam um número muito grande de instruções simples em detrimento de instruções complexas.

Comentários:

Processadores da arquitetura só utilizam LOAD e STORE para acessar a memória, depois só realizam operações envolvendo dados que estão nos registradores. Portanto, a **alternativa D está correta e é o gabarito da questão**.

16. (INAZ do Pará/CORE-SP - 2019) “O projeto do Conjunto de Instruções inicia com a escolha de uma entre duas abordagens, a abordagem RISC e a CISC”.

Disponível em: <http://producao.virtual.ufpb.br/books/edusantana/introducao-a-arquitetura-de-computadores-livro/livro/livro.chunked/ch04s04.html>. Acesso em: 13.12.2018.

Quais são características do paradigma RISC de projeto de CPU?

- A) São mais baratos, menos acesso à memória, conjunto de instruções simples.
- B) Objetivo de criar um hardware mais otimizado, com isso os programas tendem a ocupar menos espaço em memória.
- C) Grande número de registradores de propósito geral e os programas tendem a ocupar menos espaço em memória.
- D) Em geral usa mais memória para armazenamento de dados.
- E) Muitos modos de endereçamento, e foco no hardware.

Comentários:

São mais baratos, pois são mais simples. Ocorrem menos acessos à memória (apenas LOAD e STORE). Possui um conjunto de instruções simples, ao contrário da CISC (C = Complex). Portanto, a **alternativa A está correta e é o gabarito da questão**.

17. (Quadrix/CRA-PR - 2019) Possuir um conjunto de instruções simples e limitado é uma das principais características da arquitetura CISC.

Comentários:

Simple = RISC! "Complexo" = CISC! Portanto, a questão está **errada**.

18. (Quadrix/CRA-PR - 2019) A característica que mais se destaca na arquitetura RISC é que computadores pertencentes a ela realizam milhares de instruções por ciclo.

Comentários:



Vamos relembrar as características da arquitetura RISC (*Reduced Instruction Set Computer*), sendo que destaquei três itens, os quais ajudam a responder esta questão e a seguinte:

- Possui poucas instruções e todas possuem a mesma largura;
- Execução otimizada de chamada de funções;
- **Menor quantidade de modos de endereçamento;**
- Uso intenso de *pipelining*, pois é mais fácil implementar o paralelismo quando se tem instruções de mesmo tamanho;
- **Execução rápida de cada instrução (uma por ciclo de relógio);**
- Processadores RISC não requerem microcódigos (sobra mais espaço no chip);
- **Menos acesso à memória principal, instruções que acessam a memória: LOAD e STORE (arquitetura registrador – registrador, ou seja, após buscar os dados da memória e colocá-los em registradores, as operações são realizadas);**
- Maior quantidade de registradores, justamente pelo explicado no item anterior.

Portanto, a questão está **errada**.

19.(Quadrix/CREA-GO - 2019) Uma máquina RISC, geralmente, usa um conjunto de modos de endereçamento relativamente simples e direto.

Comentários:

Como existem menos modos de endereçamento e eles possuem menos acesso à memória, temos um conjunto de modos de endereçamento mais simples e direto. Portanto, a questão está **correta**.

Linguagens de Máquina e de Montagem

Conceitos

A maioria dos programadores lida com linguagens de alto nível (Pascal, Java, entre outras). Dessa forma, muito pouco da arquitetura da máquina básica fica visível. Um limite onde o projetista de computador e o programador podem ver a mesma máquina é o conjunto de instruções de máquina.

Se um programador deseja programar em linguagem de máquina (na verdade, linguagem de montagem, como veremos em breve), ele deve ter conhecimento da estrutura dos registradores e da memória, dos tipos de dados aceitos diretamente pela máquina e do funcionamento da ULA (Unidade Lógica e Aritmética). Isso tudo porque não há mais um compilador que transforme aquele código de alto nível em um código binário específico para determinada arquitetura!

Os processadores possuem um conjunto de instruções definidas por seus fabricantes e a descrição desse conjunto permite o entendimento sobre o ele (processador). Tendo esse "manual" de instruções aceitas por um determinado processador (ou família de processador), é possível programar em "baixo nível".



Instrução de Máquina

Cada instrução de máquina deve conter as informações exigidas pelo processador para a execução. Os elementos de uma instrução de máquina são:

- **Código de operação:** especifica a operação a ser realizada (ex.: ADD, E/S). A operação é especificada por um código binário conhecido como código da operação (**opcode**);
- **Referência a operando fonte:** a operação pode envolver um ou mais operandos fontes, isto é, operandos que são entradas para a operação;
- **Referência a operando de resultado:** a operação deve produzir um resultado;
- **Referência à próxima instrução:** informa ao processador onde buscar a próxima instrução depois que a execução da instrução atual estiver completa.

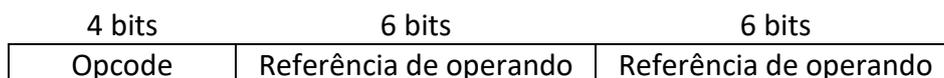
Na maioria dos casos, a próxima instrução a ser buscada vem imediatamente após a instrução corrente. Nesses casos, não há uma referência explícita à próxima instrução. Quando necessária uma referência explícita, o endereço da memória principal ou da memória virtual deve ser fornecido.

Os **operandos fonte e resultado** podem estar em uma das quatro áreas mostradas a seguir:

- **Memória principal ou virtual:** assim como as referências à próxima instrução, o endereço da memória (principal ou virtual) deve ser fornecido;
- **Registadores do processador:** um processador tipicamente possui um ou mais registradores, os quais podem ser referenciados por instruções de máquina. Cada registrador recebe um nome ou número exclusivo, e a instrução deve conter o número do registrador desejado;
- **Imediato:** o valor do operando está contido em um campo na instrução sendo executada;
- **Dispositivo de E/S:** a instrução precisa especificar o módulo e o dispositivo de E/S para a operação. Se a E/S mapeada na memória for utilizada, esse é apenas outro endereço da memória (principal ou virtual).

Representação da Instrução

Cada instrução é representada por uma sequência de bits. A instrução é dividida em campos que correspondem aos elementos que constituem a instrução. Um exemplo de instrução simples de 16 bits é:



Durante a execução da instrução, uma instrução é lida para um registrador de instrução (IR) no processador. O processador deve ser capaz de extrair os dados dos diversos campos da instrução para realizar a operação exigida.



Como é difícil para o ser humano lidar com representações binárias das instruções de máquina, uma prática comum adotada foi a de utilizar uma representação simbólica das instruções. Um exemplo dessa prática foi utilizado para o conjunto de instruções do IAS², o primeiro computador eletrônico construído pelo Instituto de Estudos Avançados de Princeton.

Os opcodes são representados por abreviações, conhecidas por mnemônicos, os quais indicam a operação. Alguns exemplos comuns são:

- ADD - Adição;
- SUB - Subtração;
- MUL - Multiplicação;
- DIV - Divisão;
- LOAD - Carrega dados da memória;
- STOR - Armazena dados na memória.

Operandos também são representados simbolicamente, como por exemplo a instrução **ADD R, X**, que pode significar a soma do valor contido na localização X com o conteúdo do registrador R.

Modos de Endereçamento

O(s) campo(s) de endereço são relativamente pequenos. Para tornar possível referenciar um grande intervalo de locais da memória principal (ou memória virtual, em alguns sistemas), uma variedade de técnicas de endereçamento foi empregada. Vamos analisar as técnicas ou modos de endereçamento mais comuns. Vamos adotar a seguinte notação:

- A (do inglês, *Address*) = conteúdo de um campo de endereço dentro da instrução;
- R = conteúdo de um campo de endereço dentro da instrução que se refere a um registrador;
- EA (do inglês, *Effective Address*) = endereço real (efetivo) do local que contém o operando referenciado;
- (X) = conteúdos do local de memória X ou do registrador X.

Endereçamento imediato: é a forma mais simples de endereçamento, no qual o valor do operando está presente na instrução:

Operando = VALOR

Esse modo pode ser utilizado para definir e utilizar constantes ou definir valores iniciais das variáveis. A vantagem é que nenhuma referência de memória (além de obter a instrução em si) é necessária para obter o operando. Isso economiza um ciclo de memória ou de cache dentro do ciclo de instrução. A desvantagem é que o tamanho do número é limitado ao tamanho do campo de endereço (geralmente pequeno, se comparado ao tamanho da palavra, que é o tamanho utilizado quando se busca um dado da memória).

² O artigo que descreve o projeto do computador IAS foi editado por nada menos que John von Neumann (nome conhecido, não?), um professor de matemática da Universidade de Princeton e do Instituto de Estudos Avançados.



Código de Operação	Referência imediata ao operando (valor do dado) Referência ao operando
---------------------------	---

Um exemplo que utiliza o modo de endereçamento imediato para mover o valor 20_{16} para o registrador B:
MOV B, #20H

Endereçamento direto: o campo de endereço possui o endereço efetivo do operando:

$$EA = A$$

Essa técnica era comum nas primeiras gerações de computadores, requer apenas uma referência à memória e nenhum cálculo especial. A limitação é que ela oferece um espaço de endereçamento limitado.

Endereçamento indireto: no endereçamento direto, o tamanho do campo de endereço geralmente é menor do que o tamanho da palavra, o que limita o espaço de endereços. A solução é ter um campo de endereço fazendo referência ao endereço de uma palavra na memória, a qual possui o endereço completo do operando:

$$EA = (A)$$

Como definido lá nas notações, os parênteses são interpretados como "conteúdo de". A vantagem principal é que, para um tamanho N de uma palavra, um espaço de endereçamento de 2^N ficará disponível. A desvantagem é que a execução da instrução requer duas referências à memória para obter o operando, uma para obter apenas o endereço e a outra para obter o operando (valor) em si.

Endereçamento por registradores: semelhante ao endereçamento direto, porém o campo de endereço faz referência a um registrador em vez de um endereço de memória:

$$EA = R$$

Por exemplo, se o conteúdo de um campo de endereço de registrador for 3, então o registrador R3 é o endereço pretendido e o valor do operando estará em R3. As vantagens são que apenas um pequeno campo de endereço é necessário e nenhuma referência é feita à memória para buscar o operando. A desvantagem é o espaço de endereçamento muito limitado, afinal não existem tantos registradores, se comparado à memória principal.

Endereçamento indireto por registradores: análogo ao endereçamento indireto, sendo que a diferença é que no lugar de referência à memória, existe referência a um registrador:

$$EA = (R)$$

As vantagens e desvantagens são basicamente as mesmas do endereçamento indireto. O endereçamento indireto por registradores utiliza uma referência à memória a menos do que o endereçamento indireto.

Endereçamento por deslocamento: combina as capacidades do endereçamento direto e do endereçamento indireto por registradores:



$$EA = A + (R)$$

Esse modo de endereçamento requer que a instrução tenha dois campos de endereço, dos quais ao menos um seja explícito. O valor contido em um campo de endereço (valor = A) é utilizado diretamente e o outro campo de endereço refere-se a um registrador cujos conteúdos são adicionados a A para produzir um endereço efetivo. Vamos ver três dos usos mais comuns a seguir.

- **Endereçamento relativo:** o endereçamento é relativo ao registrador PC (*Program Counter*), ou seja, o endereço da próxima instrução é adicionado ao campo de endereço para produzir EA. Em geral, o campo de endereço é tratado como um número complementar para essa operação. Assim, o endereço efetivo é o deslocamento relativo ao endereço da instrução;
- **Endereçamento por registrador base:** o registrador base contém um endereço da memória principal e o campo de endereço contém um deslocamento desse endereço (geralmente um inteiro sem sinal);
- **Indexação:** o campo de endereço faz referência a um endereço da memória principal e o registrador referenciado contém um deslocamento positivo desse endereço.

Endereçamento de pilha: itens são adicionados e retirados do topo da pilha, sendo que há um ponteiro cujo valor é o endereço do topo. O ponteiro da pilha é mantido em um registrador. O modo de endereçamento de pilha é uma forma de endereçamento implícito, sendo que as instruções de máquina não necessitam incluir uma referência de memória, devem apenas operar no topo da pilha.

Linguagem de Montagem

Como já vimos, um processador (CPU) entende e executa instruções de máquina (opcodes e operandos, uma sequência de zeros e uns). Uma programação direto em linguagem de máquina seria muito complexo para um programador, a não ser que ele seja um robô, porque um ser humano prefere "palavras"!

Para melhorar um pouco a vida do programador (não muito, pois aí seria o caso de linguagens de alto nível), nomes simbólicos (mnemônicos) podem ser utilizados no lugar de cada instrução binária, mas ainda há o problema de utilizar endereços absolutos, afinal de contas quem sabe os endereços fixos que pode carregar os dados? E depois, para alterar alguma coisa, seria algo bem trabalhoso! Melhorando um pouco mais, há um sistema que utiliza endereços simbólicos, onde nós queremos chegar...a **linguagem de montagem (Assembly)**, aquela que pode ser traduzidas para linguagem de máquina através de um montador (*assembler*).

Em geral, cada instrução da linguagem de montagem é traduzida em uma instrução de máquina pelo montador (1:1). É uma linguagem **dependente do hardware**, ou seja, há uma linguagem de montagem diferente para cada tipo de processador. Um programador experiente em Assembly para a arquitetura RISC terá que aprender muita coisa para programar em Assembly para CISC, por exemplo (claro que a lógica ele não perde, mas as instruções são bem diferentes). Os quatro **elementos** da linguagem de montagem são mostrados a seguir.

Comentário (opcional): pode ser colocado no lado direito de um comando ou pode ocupar uma linha inteira. Em geral, o caractere especial que especifica que a partir dali trata-se de um comentário é o ponto e vírgula (;).



Rótulo (opcional): se estiver presente, o montador define o rótulo como equivalente ao endereço no qual o primeiro byte do código objeto gerado para essa instrução será carregado. O programador pode usar o rótulo como um endereço ou como dado no campo de endereço de outra instrução. Os rótulos são utilizados com mais frequência em instruções de desvio. Abaixo podemos ver um exemplo (rótulo foi denominado **L1** e os comentários são colocados após o ponto e vírgula).

L1: SUB EAX, EDX ; subtrai conteúdo do reg EDX do conteúdo de EAX e armazena o result em EAX
JG L1 ; salta para L1 se o resultado da subtração for positivo

Operando(s): uma sentença de linguagem de montagem inclui zero ou mais operandos. Cada operando identifica um valor imediato, um registrador ou uma posição de memória. Para endereçamento por registrador, o nome do registrador é usado, ex.: MOV ECX, EBX. O endereçamento imediato indica que o valor é codificado dentro da instrução, ex.: MOV EAX, 100H. O endereçamento direto refere-se a uma posição de memória e é expresso como um deslocamento a partir do registrador de segmento DS. Exemplo:

MOV AX, 1234H ; AX <- 1234 (hexadecimal)

MOV [3420H], AX ; conteúdo de AX é movido para o endereço lógico DS:3420H

Mnemônico: nome da operação ou função da sentença da linguagem de montagem. Uma sentença pode corresponder a uma instrução de máquina, uma diretiva do montador ou uma macro. No caso de uma instrução de máquina, um mnemônico é o nome simbólico associado com um determinado opcode. Vamos ver alguns mnemônicos, mas antes é importante conhecer os registradores de uso geral (arquitetura x86, a mais cobrada em concursos). Os oito registradores de uso geral são:

- EAX: acumulador, usado em operações aritméticas;
- ECX: contador, usado em *loops*;
- EDX: registrador de dados, usado em operações de entrada/saída e em multiplicações e divisões. É também uma extensão do acumulador;
- EBX: base, usado para apontar para dados no segmento DS (*data segment*);
- ESP: apontador da pilha (*Stack Pointer*), aponta para o topo da pilha (endereço mais baixo dos elementos da pilha);
- EBP: apontador da base do *frame*, usado para acessar argumentos de procedimentos passados pela pilha;
- ESI: índice da fonte de dados a copiar (*Source Index*), aponta para dados a copiar para DS:EDI;
- EDI: índice do destino de dados a copiar (*Destination Index*), aponta para o destino dos dados a copiar de DS:ESI.

Esses oito registradores possuem tamanho de 32 bits são "estendidos". Os 16 bits de ordem mais baixa de cada um dos registradores podem ser acessados através das versões não estendidas. As versões de 16 bits possuem os mesmos nomes que versões de 32 bits, com exceção de a letra E ser retirada (ex: EAX → AX). As versões estendidas dos registradores não existem em gerações anteriores à 80386 (primeira geração de processadores 32 bits da arquitetura x86).

As versões não estendidas dos quatro primeiros registradores de uso geral dividem-se ainda em dois grupos de 8 bits cada um. O octeto (byte) de ordem mais alta é acessado trocando o X por um H (exemplo: AX →



AH), e o octeto de ordem mais baixa trocando o X por um L (ex: AX → AL). Fica fácil lembrar quando sabemos que H = High (Alta) e L = Low (Baixa).

Agora vamos ver alguns dos mnemônicos mais conhecidos e seus significados:

- mov: move dados;
- add: adição aritmética;
- sub: subtração aritmética;
- push: empilha;
- pop: desempilha;
- jmp: salto incondicional;
- int: interrupção;
- call: chamada.

Alguns exemplos de como podem ser utilizados:

```
ADD AL, BL ; AL <- AL + BL (AL e BL são os registradores com valores)
```

```
SUB AL, BL ; AL <- AL - BL
```

```
MOV AL, 1 ; AL = 1 (move 1 para AL)
```

Por fim, vamos ver um exemplo de programa completo em Assembly, o clássico "Hello World!":

```
section .data
    msg db "Hello World!",0x0a ; string "Hello world!"
    len equ $-msg ; calcula o tamanho da string msg
section .text
    ; início da seção de texto
    global _start ; onde deve começar a execução (assim como a main da ling. C)

_start: ; label start - a execução começa nesse ponto
    ; write
    mov ebx, 1 ; arquivo de saída - stdin
    mov ecx, msg ; apontador para o buffer
    mov edx, len ; tamanho do buffer
    mov eax, 4 ; chamada write ao sistema
    int 0x80 ; chamada de sistema para o kernel

    ; exit
    mov eax, 1 ; move o valor 1 para o registo eax
    mov ebx, 0 ; move o valor 0 para o registo ebx
    int 0x80 ; chamada de sistema para o kernel
```



Questões Comentadas

20. (VUNESP/Prefeitura de Ribeirão Preto-SP - 2018) Uma arquitetura de computador hipotética utiliza um microprocessador que possui instruções com o modo de endereçamento “endereçamento indireto por registrador”. Considere a instrução de máquina a seguir, que utiliza esse tipo de endereçamento, envolvendo o registrador R1.

ADD A,(R1), 8

Considerando esse contexto, e que A representa o acumulador, 8 representa um valor imediato e ADD é o mnemônico de uma instrução de máquina que realiza a operação soma, assinale a alternativa que apresenta uma funcionalidade coerente para essa instrução e que utiliza o endereçamento indireto por registrador.

- A) O resultado da soma do valor 8 com o valor do acumulador é armazenado no próprio acumulador.
- B) O resultado da soma do valor que está em R1 com o valor do acumulador é armazenado no próprio acumulador.
- C) O resultado da soma do valor 8 com o valor que está em R1 é armazenado no acumulador.
- D) O resultado da soma do valor 8 com o dado que está na memória em um endereço apontado por R1 é armazenado no acumulador.
- E) O resultado da soma do valor que está armazenado em R1 com o dado que está na memória de endereço 8 é armazenado no acumulador.

Comentários:

Quando há dois operandos, o resultado da soma (ADD) dos dois é colocado no primeiro operando. Mas a questão nos traz três, então a soma do segundo e do terceiro é colocada no primeiro (A).

Como R1 está entre parênteses (ou poderia ser colchetes também, dependendo do processador), indica que utiliza endereçamento indireto, ou seja, faz referência ao endereço de uma palavra na memória, a qual possui o endereço completo do operando.

O terceiro operando é o valor 8, simplesmente.

De tudo isso, podemos ler da seguinte forma a instrução ADD A,(R1), 8: o valor 8 é somado com o dado que está na memória em um endereço apontado por R1. O resultado é armazenado no acumulador (denominado A, pela questão).

Logo, a **alternativa D** está correta e é o gabarito da questão.



21.(FADESP/IF-PA - 2018) Em um sistema de computação, o modo mais simples de uma instrução especificar um operando é a parte da instrução referente ao endereço conter o operando de fato em vez de um endereço que descreva onde ele está. Ou seja, o operando é automaticamente buscado na memória, ao mesmo tempo que a própria instrução. Esse modo de endereçamento é denominado

- A) imediato.
- B) direto.
- C) direto via registrador.
- D) indireto.
- E) indexado.

Comentários:

Endereçamento imediato: é a forma mais simples de endereçamento, no qual o valor do operando está presente na instrução:

$$\text{Operando} = \text{VALOR}$$

Esse modo pode ser utilizado para definir e utilizar constantes ou definir valores iniciais das variáveis. A vantagem é que nenhuma referência de memória (além de obter a instrução em si) é necessária para obter o operando. Isso economiza um ciclo de memória ou de cache dentro do ciclo de instrução. A desvantagem é que o tamanho do número é limitado ao tamanho do campo de endereço (geralmente pequeno, se comparado ao tamanho da palavra, que é o tamanho utilizado quando se busca um dado da memória).

Código de Operação	Referência imediata ao operando (valor do dado) Referência ao operando
---------------------------	---

Um exemplo que utiliza o modo de endereçamento imediato para mover o valor 20_{16} para o registrador B:
MOV B, #20H

Logo, a **alternativa A** está correta e é o gabarito da questão.

22. (CESPE/ABIN - 2018) No método de endereçamento direto, a instrução contém o endereço da memória onde o dado está localizado.

Comentários:

Endereçamento direto: o campo de endereço possui o endereço efetivo do operando:

$$EA = A$$



Essa técnica era comum nas primeiras gerações de computadores, requer apenas uma referência à memória e nenhum cálculo especial. A limitação é que ela oferece um espaço de endereçamento limitado.

Logo, a questão está **correta**.

23.(FAURGS/BANRISUL - 2018) Assinale a alternativa que apresenta as características da instrução de movimentação “MVC PARM1,PARM2” na sua definição e execução.

- A) Move o endereço do PARM2 para o endereço do PARM1.
- B) Move o endereço do PARM1 para o endereço do PARM2.
- C) Move o conteúdo do PARM1 para o local onde está PARM2.
- D) Move o conteúdo de PARM2 para o local onde está PARM1.
- E) Move o conteúdo do PARM2 para o endereço do PARM1.

Comentários:

Curiosidade: a instrução MVC permite a movimentação de 1 a 256 caracteres de uma localização na memória para outra. Mas esse não é o foco, pois poderia ter o MOV que a questão poderia ser respondida também! O foco é nos parâmetros. Em Assembly, temos que ler de trás para frente, ou seja, a movimentação ocorre do segundo operando para o primeiro, então de PARM2 para PARM1, conforme denominações dadas pela questão. Como não tem parênteses no PARM1 nem no PARM2, não estamos falando de endereços, e sim a movimentação do conteúdo de PARM2 para o local onde está PARM1 (mesma lógica quando lidamos com registradores). Logo, a **alternativa D** está correta e é o gabarito da questão.

24.(UFRR/UFRR - 2019) Quanto mais um programador dominar uma linguagem de programação, melhor ele conseguirá se expressar no mundo da programação e mais recursos ele terá para escrever soluções para problemas computacionais via código.

(trecho retirado de: www.universidadedatecnologia.com.br, acesso em 18/06/2019)

Supondo que o texto acima tem caráter unicamente motivador, responda:

Qual das alternativas abaixo **NÃO** representa uma linguagem de programação de alto nível:

- A) C
- B) C++
- C) Assembly
- D) JAVA
- E) Visual Basic



Comentários:

Questão light, né? C, C++, Java e Visual Basic possuem comandos bem mais próximos do ser humano, são comandos em inglês em que um programador tem ideia do que ele faz e é mais tranquilo programar. São linguagens de alto nível. A linguagem de montagem (Assembly) possui mnemônicos para os opcodes da linguagem de máquina, lida diretamente com os registradores do processador, então podemos ver que não tem nada de alto nível! Logo, a **alternativa C** está correta e é o gabarito da questão.

25.(VUNESP/Câmara de Sertãozinho-SP - 2019) Em uma instrução de máquina, presente em uma arquitetura de computador, o modo direto de endereçamento é aquele em que no

- A) campo operando da instrução está indicado o dado.
- B) campo operando da instrução está indicado o endereço de memória, onde se localiza o dado.
- C) campo operando da instrução está indicado o endereço de memória, onde se localiza o endereço do dado.
- D) código de operação da instrução está indicado o dado.
- E) código de operação da instrução está indicado o endereço de memória, onde se localiza endereço do dado.

Comentários:

Endereçamento direto: o campo de endereço possui o endereço efetivo do operando: $EA = A$.

Essa técnica era comum nas primeiras gerações de computadores, requer apenas uma referência à memória e nenhum cálculo especial. A limitação é que ela oferece um espaço de endereçamento limitado.

Logo, a **alternativa B** está correta e é o gabarito da questão.

LISTA DE QUESTÕES

1. (CESPE/Polícia Federal - 2004) Para se determinar a capacidade de processamento e saber qual é o computador de melhor desempenho, é suficiente consultar a frequência do relógio (clock) do processador.
2. (MS CONCURSOS/CODENI-RJ - 2010) É o componente vital do sistema, porque, além de efetivamente realizar as ações finais, interpreta o tipo e o modo de execução de uma instrução, bem como controla quando e o que deve ser realizado pelos demais componentes, emitindo para isso sinais apropriados de controle. A descrição acima refere-se a?
 - A) Dispositivos de Entrada e Saída.
 - B) Memória Principal.



- C) Memória Secundária.
- D) Unidade Central de Processamento.

3. (AOCP/TCE-PA - 2012) Em computação CPU significa

- A) Central de Processamento Única.
- B) Único Centro de Processamento.
- C) Unidade Central de Processamento.
- D) Central da Unidade de Processamento.
- E) Centro da Unidade de Processamento.

4. (IF-PA/IF-PA - 2016) A Unidade Central de Processamento (UCP) é composta por um conjunto de componentes básicos, EXCETO:

- A) Unidade de Controle.
- B) Unidade de Entrada/Saída.
- C) Unidade de Aritmética e Lógica.
- D) Conjunto de Registradores.
- E) Chipset.

5. (CESPE/TCE-PA - 2016) Uma das funções de uma unidade central de processamento é buscar instruções de programas armazenados na memória principal, examiná-las e executá-las uma após a outra.

6. (INAZ do Pará/DPE-PR - 2017) O funcionário de uma empresa precisa adquirir um novo computador. Durante suas pesquisas, ele se interessou por um computador com a seguinte configuração dos componentes de hardware: 3,5 GHz, 4 GB, 1 TB, 64 bits. Nessa configuração,

- A) 64 bits é a taxa de transmissão da porta USB.
- B) 4 GB é a quantidade da memória ROM.
- C) 1 TB é a capacidade de memória RAM.
- D) 3,5 GHz é a velocidade do processador.



7. (FGV/AL-RO - 2018) Assinale a opção que indica os componentes de uma unidade central de processamento ou CPU (Central Processing Unit).

- A) Unidade lógica e aritmética, unidade de controle e registradores.
- B) Discos ópticos, disco rígido e drive.
- C) Scanner, plotter e dispositivos de entrada.
- D) Memória ROM, memória RAM e cache.
- E) Mouse, teclado e impressora.

8. (CESPE/IFF - 2018) A respeito da unidade central de processamento (CPU), julgue os itens que se seguem.

I A CPU, também denominada processador, tem como função controlar a operação do computador.

II Os registradores são responsáveis por oferecer armazenamento interno à CPU.

III A unidade de controle e a unidade aritmética e lógica fazem parte da CPU.

Assinale a opção correta

- A) Apenas o item I está certo.
- B) Apenas o item II está certo.
- C) Apenas os itens I e III estão certos.
- D) Apenas os itens II e III estão certos.
- E) Todos os itens estão certos.

9. (ESAF/SUSEP - 2010) Em uma Arquitetura RISC

- A) há poucos registradores.
- B) há pouco uso da técnica pipelining.
- C) as instruções possuem diversos formatos.
- D) as instruções são realizadas por microcódigo.
- E) as instruções utilizam poucos ciclos de máquina.



10.(FCC/TRE-AM - 2010) Numa máquina estruturada multinível, é o nível essencial para as máquinas CISC (Complex Instruction Set Computer), mas que inexistente nas máquinas RISC (Reduced Instruction Set Computer). Trata-se do nível

- A) do sistema operacional.
- B) de lógica digital.
- C) de microprogramação.
- D) convencional de máquina.
- E) do montador.

11.(CESPE/Correios - 2011) As instruções CISC são mais simples que as instruções RISC, por isso, os compiladores para máquinas CISC são mais complexos, visto que precisam compensar a simplificação presente nas instruções. Entretanto, se for usado pipeline, a complexidade do compilador CISC é reduzida, pois a arquitetura pipeline evita a necessidade de reordenação inteligente de instruções.

12.(VUNESP/UNESP - 2013) Um computador baseado em uma Unidade Central de Processamento do tipo RISC

- A) não faz uso de pipeline.
- B) executa cada instrução em um ciclo de relógio
- C) possui instruções de tamanho variável.
- D) possui muitos modos de endereçamento
- E) possui um grande conjunto de instruções.

13.(FUNDEP/IPSEMG - 2013) A arquitetura RISC de um computador possui as seguintes características, EXCETO:

- A) Formatos simples de instruções.
- B) Modos simples de endereçamento.
- C) Operações memória-para-memória.
- D) Uma instrução por ciclo.



14.(CESPE/Antaq - 2014) Atualmente, os fabricantes de computadores têm adotado exclusivamente a arquitetura RISC para o desenvolvimento de chips para processadores, dado o melhor desempenho dessa arquitetura em relação à arquitetura CISC.

15.(IADES/PCDF - 2016) Em relação ao projeto de máquinas RISC e CISC, assinale a alternativa correta.

- A) Dadas as características das instruções das máquinas CISC, o pipeline fica favorecido nessa arquitetura.
- B) Arquiteturas RISC normalmente realizam poucas operações de registrador para registrador, aumentando o acesso à memória cache.
- C) Programas para arquiteturas CISC sempre possuem tamanho menor que programas para arquiteturas RISC, devido à relação um para um de instruções de máquina e instruções de compilador.
- D) Arquiteturas RISC tendem a enfatizar referências aos registradores no lugar de referências à memória.
- E) Arquiteturas CISC usam um número muito grande de instruções simples em detrimento de instruções complexas.

16.(INAZ do Pará/CORE-SP - 2019) “O projeto do Conjunto de Instruções inicia com a escolha de uma entre duas abordagens, a abordagem RISC e a CISC”.

Disponível em: <http://producao.virtual.ufpb.br/books/edusantana/introducao-a-arquitetura-de-computadores-livro/livro/livro.chunked/ch04s04.html>. Acesso em: 13.12.2018.

Quais são características do paradigma RISC de projeto de CPU?

- A) São mais baratos, menos acesso à memória, conjunto de instruções simples.
- B) Objetivo de criar um hardware mais otimizado, com isso os programas tendem a ocupar menos espaço em memória.
- C) Grande número de registradores de propósito geral e os programas tendem a ocupar menos espaço em memória.
- D) Em geral usa mais memória para armazenamento de dados.
- E) Muitos modos de endereçamento, e foco no hardware.

17.(Quadrix/CRA-PR - 2019) Possuir um conjunto de instruções simples e limitado é uma das principais características da arquitetura CISC.

18.(Quadrix/CRA-PR - 2019) A característica que mais se destaca na arquitetura RISC é que computadores pertencentes a ela realizam milhares de instruções por ciclo.

19.(Quadrix/CREA-GO - 2019) Uma máquina RISC, geralmente, usa um conjunto de modos de endereçamento relativamente simples e direto.



20. (VUNESP/Prefeitura de Ribeirão Preto-SP - 2018) Uma arquitetura de computador hipotética utiliza um microprocessador que possui instruções com o modo de endereçamento “endereçamento indireto por registrador”. Considere a instrução de máquina a seguir, que utiliza esse tipo de endereçamento, envolvendo o registrador R1.

ADD A,(R1), 8

Considerando esse contexto, e que A representa o acumulador, 8 representa um valor imediato e ADD é o mnemônico de uma instrução de máquina que realiza a operação soma, assinale a alternativa que apresenta uma funcionalidade coerente para essa instrução e que utiliza o endereçamento indireto por registrador.

- A) O resultado da soma do valor 8 com o valor do acumulador é armazenado no próprio acumulador.
- B) O resultado da soma do valor que está em R1 com o valor do acumulador é armazenado no próprio acumulador.
- C) O resultado da soma do valor 8 com o valor que está em R1 é armazenado no acumulador.
- D) O resultado da soma do valor 8 com o dado que está na memória em um endereço apontado por R1 é armazenado no acumulador.
- E) O resultado da soma do valor que está armazenado em R1 com o dado que está na memória de endereço 8 é armazenado no acumulador.

21. (FADESP/IF-PA - 2018) Em um sistema de computação, o modo mais simples de uma instrução especificar um operando é a parte da instrução referente ao endereço conter o operando de fato em vez de um endereço que descreva onde ele está. Ou seja, o operando é automaticamente buscado na memória, ao mesmo tempo que a própria instrução. Esse modo de endereçamento é denominado

- A) imediato.
- B) direto.
- C) direto via registrador.
- D) indireto.
- E) indexado.

22. (CESPE/ABIN - 2018) No método de endereçamento direto, a instrução contém o endereço da memória onde o dado está localizado.



23.(FAURGS/BANRISUL - 2018) Assinale a alternativa que apresenta as características da instrução de movimentação “MVC PARM1,PARM2” na sua definição e execução.

- A) Move o endereço do PARM2 para o endereço do PARM1.
- B) Move o endereço do PARM1 para o endereço do PARM2.
- C) Move o conteúdo do PARM1 para o local onde está PARM2.
- D) Move o conteúdo de PARM2 para o local onde está PARM1.
- E) Move o conteúdo do PARM2 para o endereço do PARM1.

24.(UFRR/UFRR - 2019) Quanto mais um programador dominar uma linguagem de programação, melhor ele conseguirá se expressar no mundo da programação e mais recursos ele terá para escrever soluções para problemas computacionais via código.

(trecho retirado de: www.universidadedatecnologia.com.br, acesso em 18/06/2019)

Supondo que o texto acima tem caráter unicamente motivador, responda:

Qual das alternativas abaixo NÃO representa uma linguagem de programação de alto nível:

- A) C
- B) C++
- C) Assembly
- D) JAVA
- E) Visual Basic

25.(VUNESP/Câmara de Sertãozinho-SP - 2019) Em uma instrução de máquina, presente em uma arquitetura de computador, o modo direto de endereçamento é aquele em que no

- A) campo operando da instrução está indicado o dado.
- B) campo operando da instrução está indicado o endereço de memória, onde se localiza o dado.
- C) campo operando da instrução está indicado o endereço de memória, onde se localiza o endereço do dado.
- D) código de operação da instrução está indicado o dado.
- E) código de operação da instrução está indicado o endereço de memória, onde se localiza endereço do dado.



GABARITO

GABARITO



1. Errado
2. D
3. C
4. E
5. Certo
6. D
7. A
8. E
9. E
10. C
11. Errado
12. B
13. C
14. Errada
15. D
16. A
17. Errado
18. Errado
19. Certo
20. D
21. A
22. Certo
23. D
24. C
25. B



ESSA LEI TODO MUNDO CONHECE: PIRATARIA É CRIME.

Mas é sempre bom revisar o porquê e como você pode ser prejudicado com essa prática.



1 Professor investe seu tempo para elaborar os cursos e o site os coloca à venda.



2 Pirata divulga ilicitamente (grupos de rateio), utilizando-se do anonimato, nomes falsos ou laranjas (geralmente o pirata se anuncia como formador de "grupos solidários" de rateio que não visam lucro).



3 Pirata cria alunos fake praticando falsidade ideológica, comprando cursos do site em nome de pessoas aleatórias (usando nome, CPF, endereço e telefone de terceiros sem autorização).



4 Pirata compra, muitas vezes, clonando cartões de crédito (por vezes o sistema anti-fraude não consegue identificar o golpe a tempo).



5 Pirata fere os Termos de Uso, adultera as aulas e retira a identificação dos arquivos PDF (justamente porque a atividade é ilegal e ele não quer que seus fakes sejam identificados).



6 Pirata revende as aulas protegidas por direitos autorais, praticando concorrência desleal e em flagrante desrespeito à Lei de Direitos Autorais (Lei 9.610/98).



7 Concurseiro(a) desinformado participa de rateio, achando que nada disso está acontecendo e esperando se tornar servidor público para exigir o cumprimento das leis.



8 O professor que elaborou o curso não ganha nada, o site não recebe nada, e a pessoa que praticou todos os ilícitos anteriores (pirata) fica com o lucro.



Deixando de lado esse mar de sujeira, aproveitamos para agradecer a todos que adquirem os cursos honestamente e permitem que o site continue existindo.