

Material de Apoio Aula 1

Arquitetura de Von Neumann

O modelo básico dos computadores comerciais segue a arquitetura de Von Neumann (Figura 1). Nesta arquitetura, a central de processamento (CPU) possui um módulo de processamento e memória. O processador é dedicado a executar, uma a uma, as instruções de um programa. Os dados necessários à execução do programa são obtidos de dispositivos de entrada (teclado, disco etc) e armazenados, enquanto o programa encontra-se em execução, na memória da CPU. Uma vez terminada a execução, tipicamente o programa apresenta os resultados em algum dispositivo de saída (monitor, impressora etc).

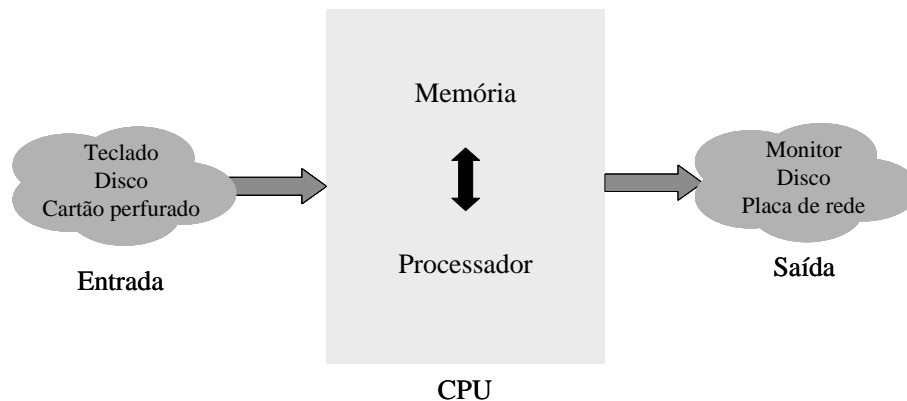


Figura 1: Arquitetura Von Neumann para arquiteturas seqüenciais.

Processo de Compilação

Um compilador é um programa (ou mesmo um conjunto de programas utilizados de forma combinada) que recebe um arquivo fonte como entrada e produz um novo arquivo como saída (Figura 2). O arquivo fonte é escrito pelo programador em uma linguagem de programação. O arquivo produzido pelo compilador é normalmente identificado como código objeto. O código objeto consiste nas instruções de baixo nível que podem ser executadas pelo computador. Normalmente o código objeto gerado por um compilador deve ser submetido a um montador (também conhecido como linkeditor), o qual pode ser executado por um computador. Observe que o código executável produzido não é portátil: diferentes compiladores/montadores são construídos para as diferentes arquiteturas de processadores – diferentes famílias de processadores possuem conjuntos distintos de instruções. É importante notar que o compilador somente gera com sucesso o arquivo objeto caso não tenha sido encontrado nenhum erro sintático no programa fonte, ou seja, não houve nenhum emprego incorreto de nenhum elemento da linguagem de programação utilizada.

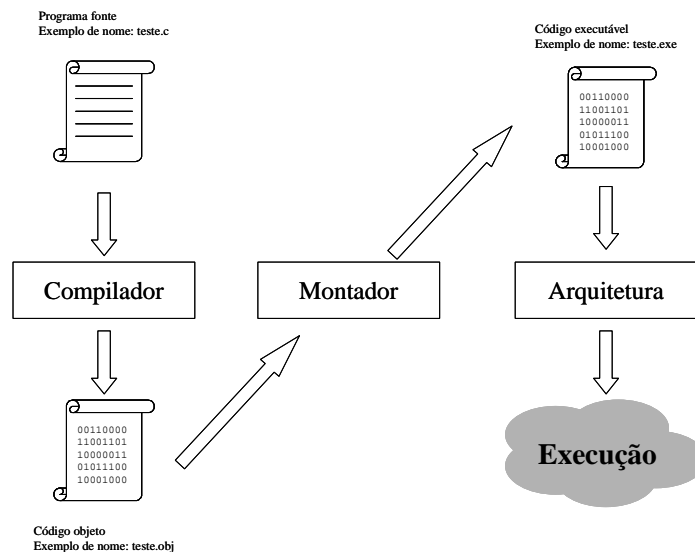


Figura 2: Esquema do processo de compilação.

Processo de Interpretação

Um interpretador é um programa que recebe como entrada um arquivo contendo um programa fonte. O interpretador é capaz de ler, linha a linha o arquivo de entrada e executar, uma a uma, as instruções que ele contém (Figura 3). Assim, um programa fonte pode ser executado em qualquer computador, bastando que exista o interpretador disponível. Note, no entanto, que a execução de um programa sobre um interpretador é mais lenta que um programa compilado, uma vez que cada linha precisa ser decodificada antes de ser executada. Assim, pode ocorrer que, caso haja alguma linha de código mal codificada (não respeitando o que foi definido na linguagem de programação utilizada), o programa termina sua execução abruptamente em erro.

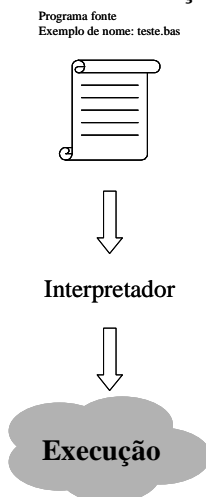


Figura 3: Esquema do processo de interpretação.

E em Java?

Java é uma linguagem compilada. No entanto, seu compilador não gera código para um processador específico, mas sim para uma máquina virtual Java (conhecida como JVM – Java Virtual Machine), conforme esquema na Figura 4. Os programas escritos em Java são, portanto portáveis sobre todos os computadores para os quais existir uma implementação da JVM. Desta forma, Java goza tanto do benefício das linguagens compiladas, uma vez que a execução de um

programa não corre o risco de ser abortada devido a um erro de sintaxe no seu código, e das linguagens interpretadas, uma vez que executa sobre uma máquina virtual que tem um comportamento semelhante a um interpretador.

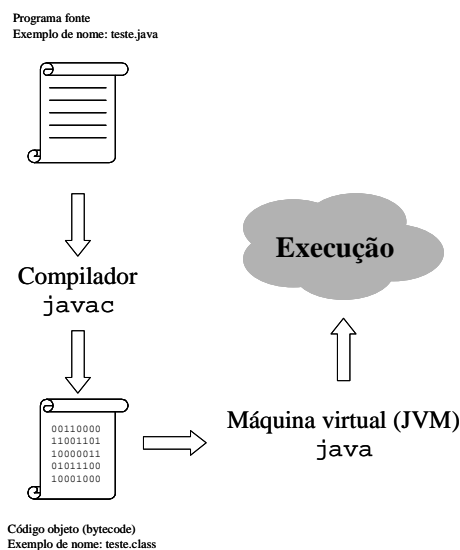


Figura 4: Esquema de compilação em Java e execução sobre uma máquina virtual.

Exercícios

1. Envie um mail para o professor contendo seu nome e seu número de matrícula. **Atenção** mande este mail através do endereço eletrônico que irás utilizar para contatar o professor durante todo o semestre. No futuro, caso necessite mudar seu endereço, comunique o professor em sala de aula.
2. A execução de um programa compilado é mais eficiente que a execução de um programa pelo processo de interpretação. Por quê?
3. O que é o **drive F**?
4. Para que servem os **diretórios** em um drive? Com que finalidade costuma-se organizar os diretórios em uma **hierarquia**?
5. Crie, no drive F um diretório chamado **lab1**. Neste diretório, você acharia interessante **criar novos sub-diretórios**? Explique; se for o caso, indique quais você criaria.
6. O que é um arquivo com a extensão **.doc**? Em qual aplicativo (Windows) é possível criar e manipular um arquivo com esta extensão?
7. Um arquivo com a extensão **.txt** é a mesma coisa que um arquivo com a extensão **.doc**? Se sim, diga porque. Se não, diga as diferenças. Em que aplicativo (Windows) é possível editar um arquivo **.txt**?
8. Abra uma **janela DOS** e execute o aplicativo **edit**. O que é uma janela DOS? O aplicativo edit oferece qual funcionalidade?

9. O que é um arquivo **.java**?

10. O que é um arquivo **.class**?

11. O que é o aplicativo **javac**?

12. O que é o aplicativo **java**?

13. Exercício sobre o ambiente Java em linha.

- a. Digite o código fonte Java apresentado abaixo em um arquivo texto e o coloque no seu diretório F:\lab1. De para este arquivo o nome de **teste.java**.

```
public class teste {
    public static void main( String args [] ) {
        String nome;
        Entrada teclado;
        Saida tela;

        nome = new String();
        tela = new Saida();
        teclado = new Entrada();

        tela.print("Digite seu nome: ");

        nome = teclado.gets();

        tela.print("Ola " + nome + ", benvindo a Laboratorio 1!!!");
    }
}
```

- b. Salve neste mesmo diretório o arquivo **es.jar** que se encontra no endereço <http://www.inf.unisinos.br/~gersonc/rep/es.jar> (identificado na página de cronograma da disciplina).
- c. Na sua janela DOS vá para o diretório F:\lab1. Digite o seguinte comando para configuração do compilador: **set CLASSPATH=.;es.jar**; e compile utilizando a seguinte linha de comando: **javac teste.java**
- d. Qual arquivo foi criado?
- e. Execute seu programa, no mesmo diretório, lançando a máquina virtual Java e seu programa através da seguinte linha de comando: **java teste**

14. O que é **GNU-Linux**?

15. Um programa escrito em Java é capaz de executar sobre GNU-Linux?

16. Descreva como você utiliza seu telefone celular (ou sua batedeira, ou seu fogão etc).