

# A MÁQUINA VIRTUAL JAVA

A Especificação da Máquina Virtual Java define o termo Máquina Virtual Java (JVM), assim:

“Uma máquina imaginária que é implementada emulando-a através de software em uma máquina real.

O código a ser executado por uma Máquina Virtual Java é armazenado em um arquivo .class. Cada arquivo .class contém o código para no máximo uma classe pública.”

O código a ser executado por uma JVM consiste de um código pré-compilado denominado byte code. A maior parte da checagem de tipos em Java ocorre em tempo de compilação, isto é, no momento em que o byte code é gerado.

Qualquer interpretador Java compatível deve ser capaz de executar qualquer programa com arquivos .class que estejam de acordo com o formato especificado para estes arquivos na Especificação da Máquina Virtual Java.

Garbage Collection (Coleta de Lixo).

Em linguagens como C / C++ o programador fica responsável por liberar toda a memória alocada ao longo da execução do programa. Programas que não desalocam memória podem eventualmente travar quando não há mais memória no sistema para alocar.

Java remove do programador a responsabilidade de liberar a memória alocada ao longo da execução de um programa. A JVM provê uma thread (executando em background) que registra toda a alocação de memória e mantém um contador do número de referências para cada ponteiro de memória. Quando a JVM encontra-se ociosa, a thread de coleta de lixo checa para ver se existem ponteiros de memória para os quais o número de referências caiu a zero. Neste caso a memória é liberada pelo coletor de lixo. A coleta de lixo ocorre automaticamente durante o tempo de vida de um programa Java e elimina a necessidade de desalocação de memória por parte do programador.

Arquivos Java são “compilados” no sentido de que são convertidos de um formato texto (como são escritos) em um conjunto de códigos (byte code) independente de máquina.

Em tempo de execução, os byte codes que formam um programa Java são carregados, verificados, e então executados pelo interpretador. O interpretador possui duas tarefas: executar o byte code Java e efetuar as chamadas apropriadas ao sistema para executar o byte code no hardware disponível. Em alguns ambientes Java, uma porção do código java verificado é compilado para código de máquina nativo e executado diretamente na plataforma do hardware. (Nestes ambientes, costumam ser compiladas as porções de código executadas com maior frequência)

Fonte://[www.marcelomoraes.com.br](http://www.marcelomoraes.com.br)