

- 1) A linguagem Java possui um diferencial que combina compilação e interpretação, visto que pode ser primeiro compilado para uma middleware chamada *bytecode* e estes são interpretados por uma máquina virtual. Não obstante, alguns termos permitiam o controle do registro das variáveis pelo programador. Este conceito tornou-se desnecessário. Uma boa estratégia de alocação depende muito de detalhes específicos no nível da arquitetura. Em Java, novamente, existe uma distinção entre estático e dinâmico, o primeiro é aplicado aos dados de uma variável em uma classe Java, onde permite que o compilador determine a localização desta variável na memória.
- 2) Um compilador analisa o código de ponta a ponta a fim de traduzi-lo, inclusive apontando todos os possíveis erros de compilação. O interpretador faz este trabalho aos poucos, linha por linha.
- 3) Compiladores são mais rápidos, citado pelo fato de traduzir o código por completo. Além disso, apresenta resultados mais confiáveis devido às várias etapas de validação e otimização. Porém, é necessário saber se determinado código é “legível” para um determinado sistema operacional, visto que podem existir funções exclusivas para uns. Já interpretadores possuem uma velocidade de execução menor, mas independe de sistemas operacionais. Por exemplo, independente do sistema operacional, basta ter Python instalado que irá rodar os códigos. Sua tipagem dinâmica também ajuda o código a possuir um alto nível.
- 4) Uma linguagem simbólica é constituída de mnemônicos, tornando as instruções mais fáceis de serem interpretadas.
- 5) Uma vantagem é o alto desempenho e velocidade, pois a linguagem C possui acesso às instruções do processador. Além disso, possui uma portabilidade para a maioria das arquiteturas atualmente.
- 6) O Assembler é o programa do sistema que traduz o código *Assembly* em linguagem de máquina (Exemplo: MARS.jar)[1]. Ele traduz cada instrução para a sequência de bits que codifica a instrução da máquina.
- 7) A tarefa do editor de ligação ou linker é reorganizar o código do programa, incorporando a ele todas as partes referenciadas no código original, resultando num código executável pelo processador. Essa tarefa pode ser realizada também pelos chamados carregadores.

[1] Programa utilizado na disciplina de Arquitetura para aprender Assembly, através de uma interface gráfica.