4.11

1 Devemos levar em consideração as convenções de nomenclatura para deixar nosso código o mais legível e documentável possível, pois um dos objetivos da programação orientada a objetos é o reaproveitamento do código e é muito importante seguir uma padronização ao escrever códigos, pois muitas vezes dois ou mais desenvolvedores trabalham no mesmo código e isso facilita e muito a leitura e o desenvolvimento posterior.

Os **atributos** (variáveis) podem começar com qualquer letra e os caracteres $ ou \_, porém não podem começar com números.

Caso o nome de um atributo (variável) seja composto por mais de uma palavra, a primeira letra de cada palavra deve ser em maiúscula.

Os **métodos** devem começar com letra maiúscula e para cada palavra, a primeira letra também deve ser maiúscula.

3 Não é necessário. Mas quando há, em um método, uma variável com o mesmo nome de um atributo. Nesse caso se você quiser acessar o atributo deve utilizar o this aumentando a legibilidade do seu código. Deixando claro que você está acessando um atributo e não uma variável local

4 A Unified Modeling Language (UML) é uma linguagem de modelagem não proprietária de terceira geração. A UML não é uma metodologia de desenvolvimento, o que significa que ela não diz para você o que fazer primeiro e em seguida ou como projetar seu sistema, mas ela lhe auxilia a visualizar seu desenho e a comunicação entre objetos.

4.13

2 A abordagem recursiva tende a ser mais rápida que a abordagem iterativa, porem, custa mais memória uma vez que precisa guardar o estado de cada chamada recursiva anterior, principalmente se a linguagem nao tem um esquema de, digamos, cache para isso. Geralmente, se você sabe que não será usado um numero muito grande de chamadas recursivas, preferia recursividade senão laços de repetição. Geralmente na abordagem recursiva longos calculos geram estouro de memória.