

# Proteção e Integridade dos Dados

Programação Orientada a Objetos



educação a distância

**UFSCar**  
virtual

# Programação Orientada a Objetos



Em um comprimido (cápsula), o conteúdo (remédio) está protegido do contato com o meio ambiente.

Em algumas definições de dicionário, encapsular significa proteger em uma cápsula.

# Programação Orientada a Objetos



Assim, os dados de um objeto devem ser protegidos.



Um relógio, por exemplo, deve marcar horas entre 0 e 23. Permitir que um relógio marque horas além disso é errado.

# Programação Orientada a Objetos

*Assim, a operação que você viu na aula anterior:*

*Massa.cor = “vermelho”;*

*não deve ser realizada no código principal.*

# Programação Orientada a Objetos



# Programação Orientada a Objetos

CLASSE CARRO  
INFORMAÇÕES PROTEGIDAS  
COR  
VELOCIDADE MÁXIMA  
PESO  
INFORMAÇÕES DE LIVRE ACESSO  
ANDAR  
CORRER  
PARAR  
TROCAR PILHA

# Programação Orientada a Objetos



As informações protegidas não podem ser acessadas de qualquer lugar. Somente os métodos que estiverem na área de livre acesso poderão modificar as informações protegidas.

Métodos de outras classes poderão acessar as informações de acesso restrito (protegidas) da classe carro somente se forem classes amigas.

# Programação Orientada a Objetos



- Acesso aos dados ocorre através do operador .
- Se for um ponteiro para um objeto, então o acesso será através do operador →



# Programação Orientada a Objetos



- Exemplos:
  - Acesso correto (dado de livre acesso):
    - Massa.correr();
  - Acesso incorreto (dado protegido):
    - Massa.cor = “vermelho”;
    - hora\_jantar = 28;

# Programação Orientada a Objetos



- Algumas características de POO
  - Proteção; ←
  - Integridade dos dados; ←
  - Herança;
  - Generalização de operações;
  - Polimorfismo.