# ACH 2003 - Computação Orientada a Objetos

Princípios SOLID

Prof. Flávio Luiz Coutinho flcoutinho@usp.br

Coleção de princípios de desenvolvimento de *software* orientado a objetos.

Coleção de princípios de desenvolvimento de *software* orientado a objetos.

Visam principalmente minimizar o custo de mudanças.

Coleção de princípios de desenvolvimento de software orientado a objetos.

Visam principalmente minimizar o custo de mudanças.

Projetos mais flexíveis, mais simples de manter e estender.

SOLID

Single responsibility principle - Princípio da responsabilidade única

SOLID

Single responsibility principle - Princípio da responsabilidade única

Uma classe deve ter apenas uma razão para mudar

SOLID

Open/closed principle - Princípio aberto/fechado

SOLID

Open/closed principle - Princípio aberto/fechado

Classes devem ser abertas para extensão, mas fechadas para modificação

SOLID

Liskov substitution principle - Princípio da substituição de Liskov

SOLID

Liskov substitution principle - Princípio da substituição de Liskov

Um método que age sobre uma instância de tipo T deve também poder agir sobre qualquer subtipo de T

SOLID

Interface segregation principle - Princípio da segregação de interface

SOLID

Interface segregation principle - Princípio da segregação de interface

Várias interfaces específicas são melhores do que uma única interface de propósito geral

SOLID

Dependency inversion principle - Princípio da inversão de dependência

SOLID

Dependency inversion principle - Princípio da inversão de dependência

Dependa de abstrações e não de implementações concretas

Exemplos...

- 1) Classe FilterSet
- 2) Classes Retângulo / Quadrado
- 3) Personagens

- (open/closed, dependency inversion)
- (Liskov)
- (Liskov, interface segregation)