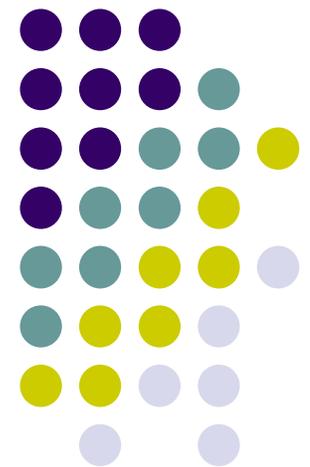
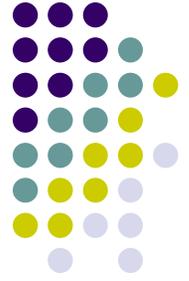


# Padrões de Software e Refatoração

---

Prof. Alberto Costa Neto  
DComp/UFS  
alberto@ufs.br





# Objetivos

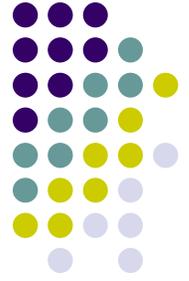
- Aprender o processo de Refatoração
- Entender Padrões de Projeto
- Realizar práticas de laboratório com ferramentas



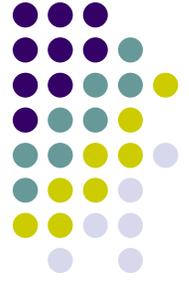
# O que é Refatoração?

- “Processo de alteração de um sistema de software de modo que o comportamento externo do código não mude, mas que sua estrutura interna seja melhorada” [Fowler]
- Problemas comuns de projeto
  - Código duplicado, obscuro e complicado

# O que é um Padrão de Software?

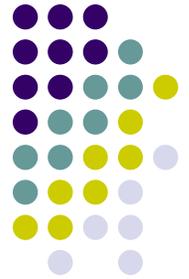


- Descrição de um problema recorrente junto com sua solução
- Obtidos através da experiência
- Catálogos sistemáticos com nomes, explicações e avaliações



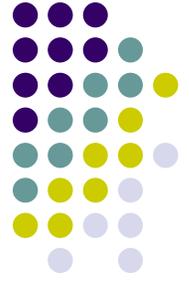
# Padrões e Refatorações

- Estão intimamente relacionados
- Direção das Refatorações
  - até um padrão
  - na direção de um padrão
  - contrária a um padrão



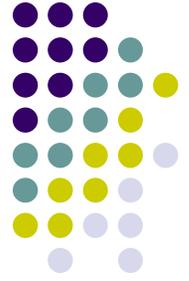
# Testes de Unidade

- Pré-condição para refatoração
- Melhora a produtividade
- Desenvolvimento dirigido por testes
  - Escreva primeiro o teste, depois o código
- Ferramentas ajudam a automatizar o processo de testes



# Resumo do Curso

- Revisão de Fatores de Qualidade do Software
- Refatorações
  - Princípios
  - Maus Cheiros
  - Catálogo
  - Prática
- Padrões de Projeto
  - Definição
  - Benefícios
  - Catálogo
  - Prática



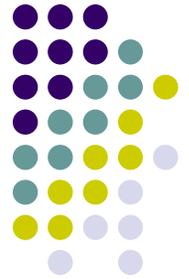
# Metodologia

- Aulas expositivas teóricas
- Seminários
- Aulas práticas
  - Expositivas com Eclipse
  - Exercícios de fixação
- Projeto



# Bibliografia Fundamental

- Refatoração: Aperfeiçoando o Projeto de Código Existente. M. Fowler.
- Design Patterns: Elements of Reusable Object Oriented Software. E. Gamma, R. Helm, R. Johnson, J. Vlissides (Gang of Four - **GoF**).
- Refactoring to Patterns. J. Kerievsky
- Test Driven Development. K. Beck



# Bibliografia de Referência

- Pragmatic Unit Testing in Java with JUnit. A Hunt, D. Thomas
- Utilizando UML e padrões. C Larman.
- Object Oriented Software Construction. 2nd. Ed. B. Meyer.
- Object-Oriented Analysis and Design. G. Booch.