



# Estimativa

**UNIP - Araraquara**

**Curso: Ciências da Computação**

**Disciplina: Qualidade de Software**

**Profº: João Paulo Moreira dos Santos**



# ADMINISTRAÇÃO DE PROJETOS

- O processo de administração de projetos de software inicia-se com um conjunto de atividades que são coletivamente denominadas **Planejamento de Projetos**.
- A primeira dessas atividades é a realização de **estimativas**.



# ADMINISTRAÇÃO DE PROJETOS

- Qual característica é a mais importante num gerente de projetos?
- A capacidade para saber o que acontecerá de errado antes que isso de fato aconteça.



# ADMINISTRAÇÃO DE PROJETOS

- A estimativa dos recursos, custo e programação de atividades para um esforço de desenvolvimento de software exige:
  - Experiência.
  - Acesso a boas informações históricas.
  - Coragem para se comprometer com medidas quantitativas.



# ADMINISTRAÇÃO DE PROJETOS

- Dilema inicial de um projeto de desenvolvimento.
  - São exigidas estimativas quantitativas, mas não existem informações sólidas disponíveis.
  - Uma análise detalhada dos requisitos de software forneceria as informações necessárias, mas a análise demora semanas ou meses para ser concluída.



# ADMINISTRAÇÃO DE PROJETOS

- Qual é o objetivo do planejamento de projetos?
  - Fornecer uma estrutura que possibilite ao gerente fazer estimativas de recursos, custos e prazos.
    - Realizada no início de um projeto.
    - Devem ser atualizadas com o tempo.



# MODELOS EMPÍRICOS DE ESTIMATIVA

- Modelo de estimativa usa fórmulas empiricamente derivadas para prognosticar informações de planejamento do projeto.
- Os dados empíricos que sustentam a maioria dos modelos derivam de uma amostra limitada de projetos.
- Nenhum modelo de estimativa é apropriado para todas as classes de software e em todos os ambientes de desenvolvimento.

# MODELOS EMPÍRICOS DE ESTIMATIVA

- Modelos de recursos consistem de uma ou mais equações empiricamente derivadas que prognosticam o:
  - esforço (em pessoas-mês);
  - duração do projeto (em meses cronológicos);



# MODELOS EMPÍRICOS DE ESTIMATIVA

- Modelos de recursos são classificados em quatro classes:
  - Modelos estáticos de variável simples;
  - Modelos estáticos de múltiplas variáveis;
  - Modelos dinâmicos de múltiplas variáveis; e
  - Modelos teóricos.

# MODELOS EMPÍRICOS DE ESTIMATIVA

- Modelo estático de variável simples:
  - $\text{Recurso} = c_1 \times (\text{características estimadas})^{c_2}$
- Modelo de Custo Construtivo.



# COCOMO

- Constructive Cost Model
  - Proposto por Barry Boehm em 1981.
  - É um modelo de estimativa de software.
    - Esforço
    - Prazo
    - Custo
  - COCOMO II: University of Southern California.



# COCOMO

## ■ Hierarquia do modelo COCOMO

- **Básico:** modelo estático de valor simples que computa o esforço (e custo) de desenvolvimento de software com uma função do tamanho de programa expresso em linhas de código estimadas.
- **Intermediário:** computa o esforço de desenvolvimento de software como uma função do tamanho do programa e de um conjunto de “direcionadores de custo” que incluem avaliações subjetivas do produto, do hardware, do pessoal e dos atributos do projeto.
- **Avançado:** incorpora todas as características da versão intermediária, com uma avaliação do impacto dos direcionadores de custo sobre cada passo (análise, projeto, etc.) do processo de engenharia de software.



# COCOMO

- O COCOMO considera três modos de desenvolvimento:
  - **Modo orgânico:** projetos simples, relativamente pequenos, nos quais pequenas equipes com boa experiência em aplicações trabalham num conjunto de requisitos não tão rígidos.
  - **Modo semidestacado:** um projeto intermediário (em tamanho e complexidade) no qual equipes com níveis de experiência mistos devem atingir uma combinação de requisitos rígidos e não tão rígidos.
  - **Modo embutido:** um projeto de software que deve ser desenvolvido dentro de um conjunto rígido de restrições operacionais, de hardware e de software.

# COCOMO

- As equações COCOMO básicas assumem a forma:
- $E = a_b(\text{KLOC}) \exp(b_b)$
- $D = c_b(E \exp(d_b))$

# COCOMO

## ► Tabela COCOMO básico

Projeto de software	$a_b$	$b_b$	$c_b$	$d_b$
Orgânico	2,4	1,05	2,5	0,38
Semidestacado	3,0	1,12	2,5	0,35
Embutido	3,6	1,20	2,5	0,32



# COCOMO

- Atributos direcionadores do custo
- **1- Atributos do produto**
  - Confiabilidade exigida do software.
  - Tamanho do banco de dados da aplicação.
  - Complexidade do produto.
- **2- Atributos do hardware**
  - Restrições ao desempenho de *run-time*.
  - Restrições de memória.
  - Volatilidade do ambiente de máquina virtual.
  - Tempo de *turnaround* (tempo para completar o ciclo) exigido.





# COCOMO

- Atributos direcionadores do custo
- **3- Atributos de pessoal**
  - Capacidade de análise.
  - Capacidade em engenharia de software.
  - Experiência em aplicações.
  - Experiência em máquina virtual.
  - Experiência em linguagens de programação.
- **4- Atributos de projeto**
  - Uso de ferramentas de software.
  - Aplicação de métodos de engenharia de software.
  - Cronograma de atividades de desenvolvimento exigido.



# COCOMO

- Cada um dos quinze atributos é classificado de acordo com uma escala de 6 pontos que varia de “muito baixo” a “extremamente elevado” (em importância e valor).
- Baseando-se na classificação, um multiplicador de esforços é determinado.
  - O produto de todos os resultados de multiplicadores de esforços torna-se um **fator de ajustamento de esforço** (EAF). Variam de 0,9 a 1,4.

# COCOMO

- As equações COCOMO intermediário assume a forma:
- $E = a_i(\text{LOC}) \exp(b_i) \times \text{EAF}$

# COCOMO

➤ Tabela COCOMO intermediário

Projeto de software	$a_i$	$b_i$
Orgânico	3,2	1,05
Semidestacado	3,0	1,12
Embutido	2,8	1,20



# COCOMO

- Exemplo projeto de nível básico semidestacado:
  - $LOC = 33,3$