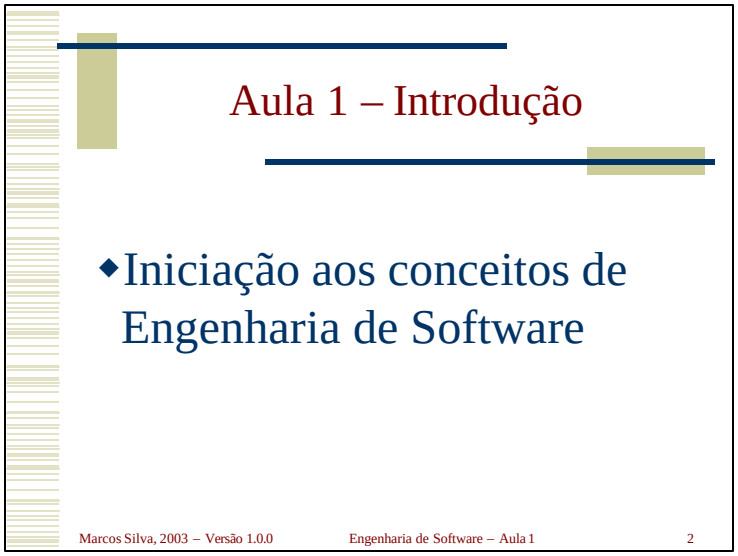
Engenharia de Software

Prof. Marcos Antonio da Silva

2003

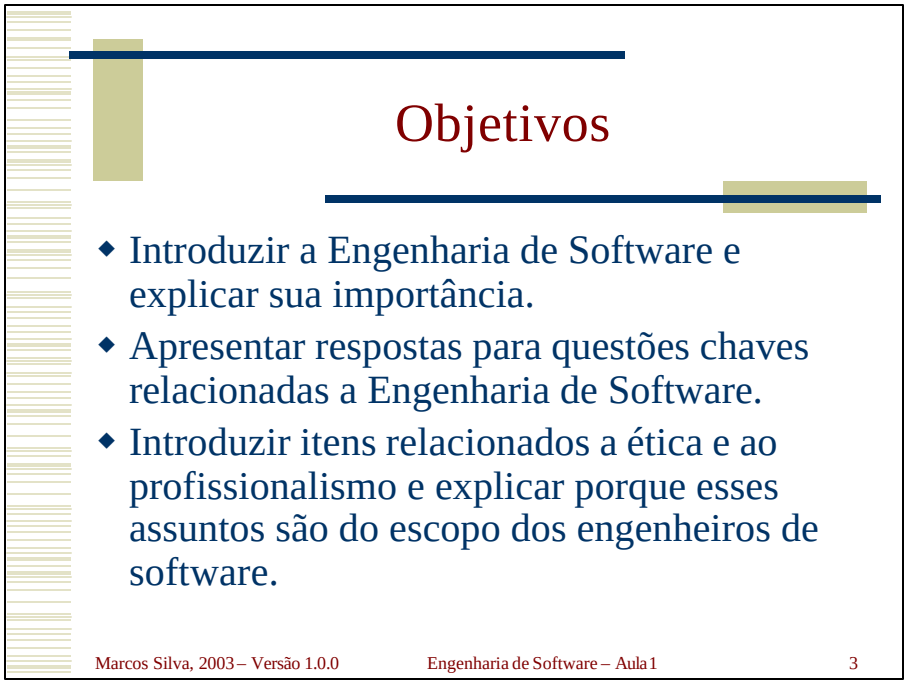
Baseado nos slides de:

©Ian Sommerville 2000 - Software Engineering, 6th edition



Aula 1

Introduz os conceitos básicos de Engenharia de Software.



Motivação da aula:

Ressaltar a importância da Engenharia de Software na produção de Software com qualidade, prazo e custo.

Assuntos abordados

- Questões mais freqüentes sobre Engenharia de Software.
- Responsabilidade Ética e Profissional.

Engenharia de Software

- A economia de TODAS as nações desenvolvidas depedem de software.
- Mais e mais sistemas são controlados por software.
- A Engenharia de Software está relacionada com teorias, métodos e ferramentas para o desenvolvimento profissional de software.
- Os gastos com Engenharia de Software representam um fração significante do PIB de todos os países desenvolvidos.

Custos de Software

- Custos de Software geralmente dominam o custo de um sistema. Num PC, esses custos em geral, são maiores que o custo do hardware.
- O Software custa mais para ser mantido do que para ser desenvolvido. Para sistemas de vida longa, a manutenção custa muitas vezes mais que o seu desenvolvimento.
- Uma das preocupações da Engenharia de Software é com realação aos custos efetivos do desenvolvimento do Software.

FAQs sobre Engenharia de Software

- O que é Software?
- O que é Engenharia de Software?
- Qual a diferença entre Engenharia de Software e Ciência da Computação?
- Qual a diferença entre Engenharia de Software e Engenharia de Sistemas?
- O que é o Processo de Software?
- O que é um modelo de processo de Software?

FAQs sobre Engenharia de Software

- Quais os custos da Engenharia de Software?
- Quais os métodos da Engenharia de Software?
- O que é CASE (Computer-Aided Software Engineering)?
- Quais as características de um bom software?
- Quais os desafios-chave que a Engenharia de Software enfrenta?

O que é Software?

- Programas de computador e sua documentação associada.
- Produtos de software podem ser desenvolvidos para um cliente em particular ou ainda para o mercado em geral (Software de Prateleira)
- Produtos de Software podem ser:
  - Genéricos – desenvolvidos para serem vendidos para uma gama de diferentes consumidores;
  - Bespoke (personalisados) – desenvolvidos para um cliente em particular de acordo com sua especificação.

O que é Engenharia de Software?

- A Engenharia de Software é uma disciplina cujo escopo está relacionado a todos os aspectos da produção de Software.
- Os engenheiros de Software devem adotar abordagens sistemáticas e organizadas no seu trabalho e utilizar ferramentas e técnicas apropriadas dependendo do problema a ser solucionado, das restrições e dos recursos disponíveis.

Qual a diferença entre Engenharia de Software e Ciência da Computação?

- A Ciência da Computação está relacionada com as teorias e fundamentos; A Engenharia de Software preocupa-se com aplicabilidade prática de desenvolvimento e entrega de software utilizável.
- Atualmente, a Ciência da Computação não é auto-suficiente para atuar como uma fornecedora de uma solução completa para a Engenharia de Software.
- A Engenharia de Software complementa o trabalho da Ciência da Computação.

Qual a diferença entre a Engenharia de Software e a Engenharia de Sistemas?

- A Engenharia de Sistemas aborda todos os aspectos de desenvolvimento de um sistema baseado em computador, incluindo hardware, software e engenharia de processos. A Engenharia de Software é apenas parte desse processo.
- Engenheiros de Sistemas estão envolvidos na especificação dos sistemas, *design* de arquitetura, integração e implantação do mesmo.

O que é Processo de Software?

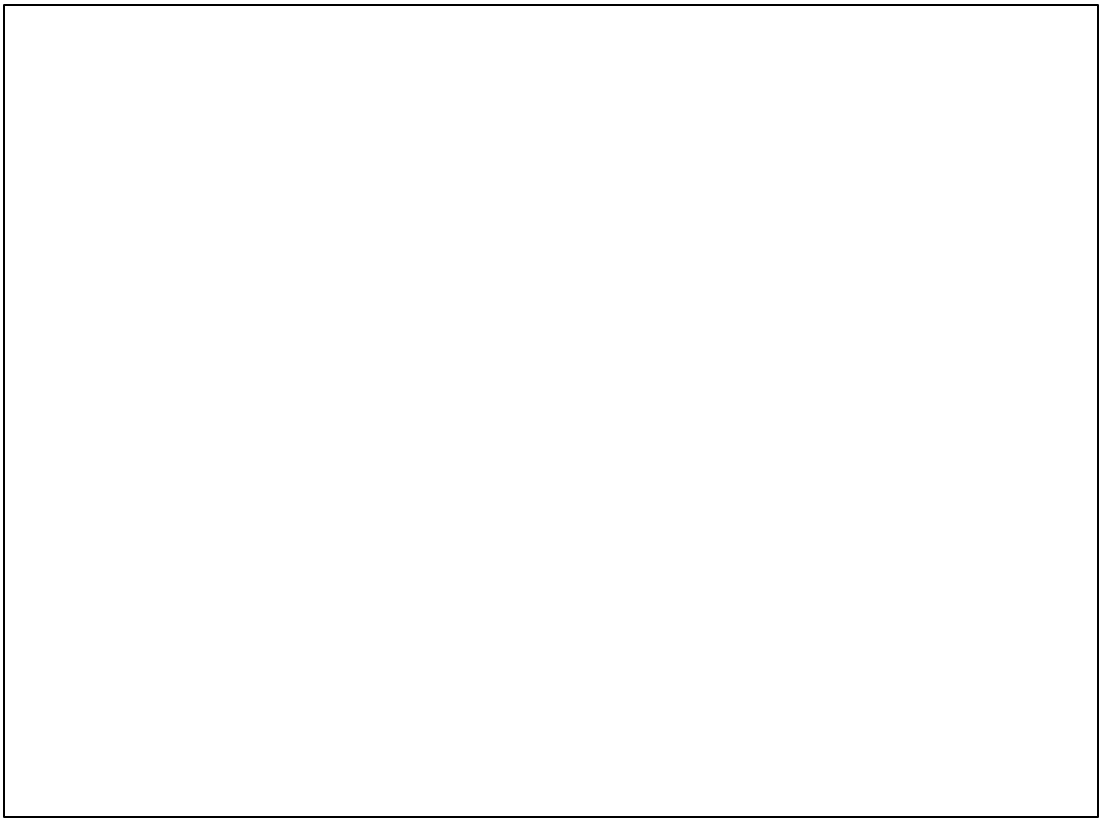
- Um conjunto de atividades cujo objetivo é o desenvolvimento ou evolução de um software.
- Algumas atividades genéricas em todo processo de desenvolvimento de software são:
  - Especificação – o que o sistema deve fazer e suas restrições de desenvolvimento;
  - Desenvolvimento – produção do sistema de software;
  - Validação – verificar se o software é o que o cliente realmente quer;
  - Evolução – modificar o software em respostas aos pedidos de mudanças.

O que é um Modelo de Processo de Software?

- Uma representação simplificada do processo de software, apresentada de uma perspectiva específica.
- Exemplos de perspectivas de processos:
  - Workflow – seqüência de atividades
  - Data-flow – fluxo de dados/informações
  - Role/action – quem faz o quê
- Modelos genéricos de processos:
  - Waterfall (Modelo *Cascata*)
  - Desenvolvimento Evolucionário
  - Transformação formal
  - Integração a partir de componentes reutilizáveis.

Quais os custos associados a Engenharia de Software?

- Aproximadamente, 60% dos custos estão relacionados ao desenvolvimento, 40% com testes. Para um software personalizado, custos de evolução em geral excedem os custos de desenvolvimento.
- Os custos variam dependendo do tipo de sistema s ser desenvolvido e dos requisitos atribuidos ao sistema, tais como performance e confiabilidade.
- Custos de distribuição dependem do modelo de desenvolvimento adotado.



Quais são os métodos de Engenharia de Software?

- Abordagens estruturadas para desenvolvimento de software que incluem modelagem de sistemas, notações, regras, recomendações de *design* e orientação de processos:
- Descrições de Modelos
  - Descrições de modelos gráficos que devem ser produzidos
- Regras
  - Restrições aplicadas aos modelos do sistema
- Recomendações
  - Recomendações sobre boas práticas de design
- Orientação de processos
  - Quais atividades seguir

O que é CASE (Computer-Aided Software Engineering)

- Sistemas de software que pretendem prover suporte para automação nas atividades do processo de desenvolvimento de software. Os sistemas CASE são usados em geral para apoiar os métodos.
- Upper-CASE (Alto nível)
  - Ferramentas para apoiar as primeiras atividades do processo, como gerenciar requisitos e design.
- Lower-CASE (Baixo nível)
  - Ferramentas para apoiar as atividades finais, como programação, *debug* e testes

Quais os atributos de um bom Sotware?

- O software deve possuir as funcionalidades e performance requeridas pelo usuário e deve ser passível de manutenção, confiável e usável.
  - Manutenibilidade
    - O Software deve evoluir para atender as necessidades de mudanças.
  - Confiabilidade
    - O Software deve ser confiável.
  - Eficiência
    - O Software não deve gastar todos os recursos do sistemas, usando apenas o necessário especificado.
  - Usabilidade
    - O Software deve ser utilizável pelos usuários para o qual foi projetado.

Quais os desafios encontrados pela Engenharia de Software?

- Manter sistemas legados, conviver com crescente diversidade e enfrentar demandas para prazos de entrega mais curtos.
  - Sistemas legados
    - Sistemas velhos e úteis devem ser mantidos e atualizados.
  - Heterogeneidade
    - Sistemas distribuídos, que incluem uma mistura de diferentes tipos de hardware.
  - Entrega
    - Há uma pressão crescente para uma entrega rápida do software.

Responsabilidade ética e profissional

- A Engenharia de Software envolve grandes responsabilidades, além de simplesmente a aplicação de conhecimentos técnicos.
- Os engenheiros de software devem comportar-se de forma honesta e ética de forma a serem respeitados como profissionais.
- Comportamento ético é muito mais que simplesmente respeitar as leis vigentes.

Itens de responsabilidade profissional

- *Confidencialidade*
  - Os engenheiros devem respeitar a confidencialidade dos de seus empregadores e/ou clientes independente de existir um acordo assinado.
- *Competência*
  - Os engenheiros não devem atuar for a do seu nível de competência. Devem apenas aceitam tarefas que estejam a ao alcance das suas capacidades e compentências.

Itens de responsabilidade profissional

- *Direitos de Propriedade Intelectual*
  - Os engenheiros deve estar a par das leis de uso de propriedade intelectual como patentes, marcas registradas, copyright, etc… Devem ter cuidado e garantir que a propriedade intelectual de empregadores e/ou clientes esteja protegida.
- *Uso indevido do conhecimento*
  - Os engenheiros de software não devem usar suas habilidades técnicas para usar indevidamente recursos computacionais não autorizados. Desde jogos até disseminação de vírus.

Código de ética da ACM/IEEE

- A comunidade profissional tem cooperado para produzir um código de ética prática.
- Membros dessas organizações assinam esse código ao se associarem as mesmas. (SBC segue o código de ética da ACM/IEEE)
- Este código contém Oito Princípios relacionados ao comportamento e decisões dos profissionais, sejam engenheiros, cientistas, professores ou mesmo alunos na carreira.

Código de Ética

Preambulo

- Uma versão resumida do código sumariza as aspirações num alto nível de abstração; as clausulas são incluídas numa versão completa, com exemplos e detalhes de como devemos atuar como profissionais da Engenharia de Software. Sem as aspirações, os detalhes tornam-se muito legalistas e tediosos; sem os detalhes as aspirações tornam-se vazias; juntas, as aspirações e os detalhes formam um código muito coeso.
- Os engenheiros de software devem comprometer-se a fazer a análise, especificação, design, desenvolvimento, teste e manutenção de um software em benefício e respeito a profissão. Em acordo com seu comprometimento com a saúde, segurança e bem-estar do público, os engenheiros de software devem aderir aos Oito Princípios, a seguir:

Código de Ética

- **Princípios**
  - 1. Públicos
    - Os Engenheiros de Software devem atuar consistentemente com o interesse público.
  - 2. Clientes e empregadores
    - Os Engenheiros de Software devem atuar de maneira que os interesses de seus empregadores e/ou clientes sejam consistentes com o interesse público.
  - 3. Produtos
    - Os Engenheiros de Software devem assegurar que os produtos e modificações relacionadas estejam de acordo o máximo possível com os padrões profissionais.

Código de Ética

- 4. Julgamento
  - Os Engenheiros de Software devem manter integridadde e independência nos seus julgamentos profissionais.
- 5. Gerência
  - Gerentes e líderes de Engenharia de Software devem adotar e promover uma abordagem ética para o gerenciamento do desenvolvimento e manutenção de software.
- 6. Profissão
  - Os Engenheiros de Software devem desenvolver a integridade e reputação da profissão de forma consistente com os interesses públicos.

Código de Ética

- 7. Companheirismo
  - Os Engenheiros de Software devem ser honestos e apoiar seus colegas.
- 8. Interesses Próprios
  - Os Engenheiros de Software devem estar envolvidos durante toda carreira com o aprendizado das práticas relacionadas relacionadas a profissão e promover a ética na prática da mesma.

Dilemas Éticos

- Desacordos nos princípios com as políticas do gerente senior.
- Atuação do empregador de uma forma não-ética, por exemplo, entregando um sistema de uma UTI sem ter testado completamente o sistema.
- Participação no desenvolvimento de armas de destruição em massa, como armas nucleares ou biológicas.

Pontos-chave

- A Engenharia de Software é uma disciplina que está relacionada com todos os aspectos da produção de software.
- Os produtos de software consistem no desenvolvimento de programas e a documentação associada. Atributos essenciais destes produtos são: manutenibilidade, confiabilidade, eficiência e usabilidade.
- O processo de software consiste de atividades que estão envolvidas no desenvolvimento dos produtos de software. Atividades básicas são: especificação, desenvolvimento, validação e evolução do software.

Pontos-chave

- Métodos são formas organizadas de produzir software. Eles incluem sugestões para o processo a ser seguido, as notações a serem usadas, regras que governam a as descrições do sistema que será produzido e linhas mestras para o design.
- Ferramentas CASE são sistemas de software que são projetados para apoiar as atividades rotineiras do processo de desenvolvimento de software, como editar digramas de design, verificar consistências e manter o controle dos casos de testes a serem executados.

Pontos-chave

- Os Engenheiros de Software tem responsabilidades para com a sua profissão e com a sociedade. Eles não devem simplesmente estar preocupados apenas com aspectos técnicos.
- Sociedades de profissionais (SBC, ACM, IEEE, etc.) publicam seus códigos de conduta que regem o padrão de comportamento esperado de seus membros.

Questões

1. Basedo nas estimativas de custos de software discutidas, explique por é apropriado considerar o software como sendo bem mais que o programa executado pelos usuários finais de um sistema.
2. Quais os quatro atributos importantes que todos os produtos de software devem ter? Sugira outros quatro atributos que podem ser significantes.
3. Qual a diferença entre um “Modelo de Processo de Software” e o “Processo de Software” em si. Sugira duas formas na qual um modelo pode ser útil na identificação de melhorias nos processos.
4. Explique porque os custos associados aos testes são particularmente altos para produtos de software genéricos que são vendidos em larga escala.

Questões

5. Os métodos de Engenharia de Software tornam-se aplamente usados quando tecnologias CASE estão disponíveis para suportá-los. Sugira cinco tipos de métodos que podem ser providos por ferramentas CASE.
6. Além dos desafios encontrados por sistemas legados, heterogeneidade e entrega rápida, identifique outros problemas e desafios que a Engenharia de Software enfrenta ou pode vir a enfrentar.
7. Discuta se os profissionais da área devem ser certificados da mesma forma que médicos ou advogados (exames dos conselhos regionais)? Em suma, a profissão deve ser regulamentada?
8. Para cada clausula do Código de Ética da ACM/IEEE, sugira exemplos que ilustrem adequadamente o item.