

# Engenharia de Software e Sistemas

Revisão para o 1º Exercício Escolar

Monitores:  
Ângelo de Sant'Ana  
José Luiz



UNIVERSIDADE  
FEDERAL  
DE PERNAMBUCO

# Questão 1

No contexto de desenvolvimento de software,  
explique e exemplifique os conceitos de:  
projeto, processo e produto.

# Projeto, Processo e Modelo de Processo

## ■ PROJETO

Um empreendimento temporário para criar um produto ou serviço único.

Características dos projetos

- *Prazo limitado*
- *Recursos limitados e definidos a priori*
- *Data estipulada da conclusão*

# Projeto, Processo e Modelo de Processo

## ■ PROCESSO

Um conjunto estruturado de atividades, práticas, artefatos e ferramentas necessários para o desenvolvimento de um sistema de software.

Exemplo de atividades:

- *Especificação*
- *Desenvolvimento*
- *Validação*
- *Evolução*

Exemplos de processos: Processo Unificado (RUP), OpenUP, Programação Extrema (XP).

# Projeto, Processo e Modelo de Processo

## ■ MODELO DE PROCESSO

- Um modelo de processo de software apresenta a descrição de um processo de uma perspectiva particular, normalmente **focando apenas em algumas atividades**.
- Um modelo é usado para **entendimento** e **comunicação** do processo, e como base para análise, execução, gerência e melhoria do processo.
- Idealmente a descrição deve ser **formal** e completa para permitir, por exemplo, automação.
- A descrição deve ser apresentada em diferentes níveis de abstração.

# Projeto, Processo e Modelo de Processo

## ■ MODELO DE PROCESSO

- Exemplos:

**Modelo cascata:** fases separadas e distintas de especificação e desenvolvimento.

**Desenvolvimento iterativo:** sistema desenvolvido através de várias etapas (iterações)

**Engenharia de software baseada em componentes:** o sistema é montado a partir de componentes existentes.

## Questão 2

Sobre o OPENUP, assinale para cada alternativa V - quando verdadeira ou F - quando falsa. Para as alternativas falsas, explique o porquê.

- ( ) A definição detalhada dos requisitos e da arquitetura do sistema é considerada um marco que indica a finalização da fase de elaboração.
- ( ) O OPENUP é um processo de desenvolvimento cascata.
- ( ) O OPENUP é um processo para desenvolvimento de software iterativo e incremental, baseado em casos de uso e centrado na arquitetura do sistema.
- ( ) Os testes de aceitação são sempre executados no ambiente dos desenvolvedores e concentram-se na fase de construção.
- ( ) O objetivo do plano de iteração é apresentar uma visão detalhada de todas as atividades, tarefas e recursos necessários durante todo o projeto.

# Ciclo de vida do projeto

( ) A definição detalhada dos requisitos e da arquitetura do sistema é considerada um marco que indica a finalização da fase de elaboração.

- **Estruturado em quatro fases**

- Conceção (define o escopo do projeto)
- Elaboração (detalha os requisitos e a arquitetura)
- Construção (desenvolve o sistema)
- Transição (implanta o sistema)

# Testes de aceitação

(F) O OPENUP é um processo de desenvolvimento *cascata*.

(V) O OPENUP é um processo para desenvolvimento de software *iterativo e incremental*, baseado em casos de uso e centrado na arquitetura do sistema.

(F) Os testes de aceitação são sempre executados no ambiente dos desenvolvedores e concentram-se na fase de construção.

**São testes funcionais, realizados pelo usuário, objetivando demonstrar a conformidade com os requisitos do software. São os testes “caixa-preta”. Pode ser de duas categorias: *alfa* (usuário testa na presença e ambiente do desenvolvedor que registra problemas) e *beta* (usuário testa no seu ambiente normal com problemas reportados ao desenvolvedor). E concentram-se na fase de construção.**

# Plano de Iteração

( ) O objetivo do plano de iteração é apresentar uma visão detalhada de todas as atividades, tarefas e recursos necessários durante todo o projeto.

## ■ Objetivo

- fornecer à equipe um lugar central para informações a respeito dos objetivos da iteração.
- plano detalhado com as atribuições das tarefas; o que deve ser entregue na iteração.
- Também ajuda a equipe a monitorar o progresso da iteração e mantém os resultados da avaliação da iteração, que podem ser úteis para melhorar a próxima iteração.
- A avaliação é feita segundo critérios de sucesso previamente definidos.

## Questão 3

Explique com suas palavras o que vem a ser uma iteração no contexto do desenvolvimento iterativo e incremental.

# Desenvolvimento Iterativo

## ■ Definição

- Desenvolvimento iterativo é uma forma de “**quebrar**” o desenvolvimento de uma grande aplicação em pequenos pedaços de software.
- Nesse processo é possível **dividir** o planejamento de desenvolvimento em uma série de períodos de desenvolvimento de duração fixa chamados **iterações**.
- As iterações são usadas para **reduzir o risco** de projetos de desenvolvimento de software e para entregar **funcionalidade de valor alto** ao usuário antecipadamente no ciclo de vida.

# Desenvolvimento Iterativo

## ■ Funcionamento

- Divide-se o projeto em subconjuntos de funcionalidade baseados, por exemplo, em **casos de uso** ou em cenários de caso de uso.
- No final de cada iteração, demonstra-se ou entrega-se estas partes de funcionalidade às partes interessadas, para que elas possam usar os recursos e fornecer **feedback**.
- Este feedback pode ser incorporado em futuras iterações.

# Desenvolvimento Iterativo

## ■ Importante

O modelo iterativo inclui mudança por meio de um *loop de feedback*, enquanto os modelos não iterativos buscam *evitar mudanças*, o que, por sua vez, *aumenta o risco* de que o sistema desenvolvido não atenda às necessidades das partes interessadas.

# Questão 4

Elabore um diagrama de casos de uso para a seguinte descrição:

*“Um sistema acadêmico requer log(in/off) para ser usado. Neste sistema, cada Professor propõe as disciplinas que deseja ofertar, indicando os dias e horários sugeridos. As disciplinas ofertadas então são confirmadas por um líder de grupo de pesquisa (também professor), o qual informa os horários definitivos para as disciplinas. Secretárias complementam cada disciplina com um código e descrição. Alunos votam em quais disciplinas pretendem cursar. Em uma data específica, o Coordenador (também professor) remove do sistema disciplinas com votação inferior a x votos. Um Administrador pode cadastrar usuários, bem como bloquear usuários já cadastrados.”*

# Questão 5

Uma equipe de desenvolvimento determinou e planejou 9 tarefas de acordo com a tabela abaixo. Crie a rede de atividades e determine o caminho crítico.

Tarefa	Duração (Dias)	Dependências
T01	10	
T02	5	
T03	7	T01 (M1)
T04	15	T02 (M2)
T05	4	
T06	8	T02, T03 (M3)
T07	4	T05 (M4)
T08	10	T03, T05 (M5)
T09	8	T04, T06, T07 (M6)

# Questão 6

Apresente o grafo do fluxo de controle para:

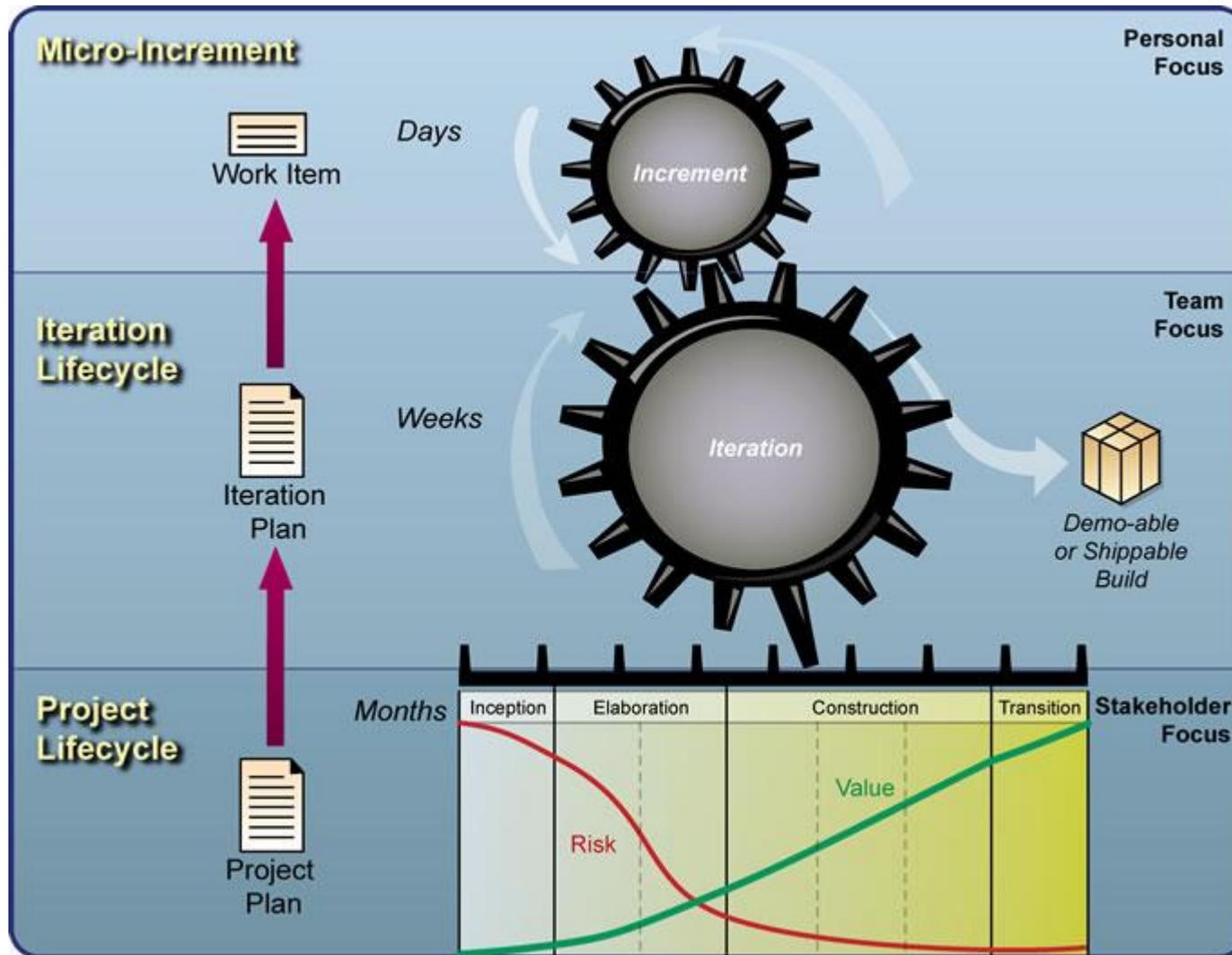
```
public static int searchSorted(int[] a, int x) {  
    if(a==null) return -1;  
    int low=0;  
    int high=a.length() - 1;  
    while(low <= high) {  
        int mid=(low+high)/2;  
        if(x==a[mid]) return mid;  
        if(x<a[mid]) high=mid - 1;  
        else low=mid + 1;  
    }  
    return -1;  
}
```

Bem como um conjunto de predicados e dados de testes necessários para exercitar todas as suas instruções.

## Questão 7

Explique o processo de desenvolvimento iterativo e incremental no contexto do OpenUP.

# Processo Iterativo e Incremental



## Questão 8

Explique a distinção entre “Requisitos de usuário” X “Requisitos de Sistema” e entre “Requisitos Funcionais” X “Requisitos Não-funcionais”.

# Requisitos

- **O que é um requisito?**
  - Pode ser um descrição abstrata de alto nível de um serviço, uma restrição de sistema ou até uma especificação matemática, entre outras coisas.
  - Define o problema cujo desenvolvimento do sistema pode resolver. O sistema tem que ser desenvolvido de modo a satisfazer todos os seus requisitos.

# Requisitos

## ■ Requisitos Funcionais

- Serviços que o sistema deve fornecer
- Como o sistema deve reagir a entradas específicas
- Como o sistema deve se comportar em determinadas situações.

# Requisitos

## ■ Requisitos Não - funcionais

- Restrições sobre serviços ou funções oferecidos pelo sistema, tais como restrições de tempo, restrições sobre o processo de desenvolvimento, padrões, linguagens de programação, etc.
- Definem propriedades.
- Podem ser mais críticos do que os funcionais. Se não satisfeitos, o sistema pode ficar inútil.

# Requisitos

## ■ Requisitos de Usuários

- Requisitos funcionais e não-funcionais descritos de modo a serem compreensíveis por usuários que não têm conhecimento técnico detalhado.  
Declarações de alto nível em linguagem natural

# Requisitos

## ■ Requisitos de Sistema

- Um documento estruturado estabelecendo descrições detalhadas das funções, serviços e restrições operacionais do sistema.  
Define o que deve ser implementado e fornece base para desenvolvimento

## Questão 9

Explique por que um excelente desenvolvedor nem sempre se torna um bom gerente de projetos de software.

## Questão 10

Por que podemos considerar o teste como uma atividade “destrutiva”? Explique.

# Teste de Software

## ■ Definição

- Teste de software é o processo de execução de um produto para determinar se ele **atingiu suas especificações e funcionou corretamente** no ambiente para o qual foi projetado.
- O seu objetivo é **revelar falhas** em um produto, para que as causas dessas falhas sejam **identificadas** e possam ser **corrigidas** pela equipe de desenvolvimento antes da entrega final.

# Teste de Software

- **Caráter Destrutivo**

Dizemos que sua natureza é “destrutiva”, e não “construtiva”, pois visa ao **aumento da confiança** de um produto através da **exposição de seus problemas**, porém antes de sua entrega ao usuário final.

# Questão 11

Explique as abordagens de testes apresentadas na disciplina.

# Testes

## ■ Teste de unidade

- Investiga a qualidade de componentes individuais (ex: métodos, classes)
- Testa comportamento (especificação) e estrutura interna (lógica e fluxo de dados)

## ■ Teste de sistema

- Investiga o funcionamento da aplicação como um todo

## ■ Teste de integração

- Testa a integração dos módulos para garantir o funcionamento do sistema final

## ■ Teste de aceitação

- Testes funcionais, realizados pelo usuário, objetivando demonstrar a conformidade com os requisitos do software

# Referências

- **Slides de aulas da disciplina Engenharia de Software e Sistemas do CIn-UFPE, disponíveis no site da disciplina**  
<https://sites.google.com/a/cin.ufpe.br/if682/>
- **O que desenvolvimento de software iterativo?**  
<http://www.rfranco.org/2013/09/o-que-e-desenvolvimento-de-software-iterativo/>
- **IBM - Desenvolvimento Iterativo**  
[http://pic.dhe.ibm.com/infocenter/clmhelp/v3r0/index.jsp?topic=%2Fcom.ibm.team.concert.doc%2Ftopics%2Fc\\_practice\\_iterative\\_development.html](http://pic.dhe.ibm.com/infocenter/clmhelp/v3r0/index.jsp?topic=%2Fcom.ibm.team.concert.doc%2Ftopics%2Fc_practice_iterative_development.html)
- **Introdução a Teste de Software**  
<http://www.devmedia.com.br/artigo-engenharia-de-software-introducao-a-teste-de-software/8035>