

Aula 1

Fundamentos de Desenvolvimento de Software

Profª Luciane Yanase Hirabara Kanashiro

1

Conversa Inicial

2

Software

- Com certeza você já ouviu falar de software...
- Mas saberia definir o que é software? O que diferencia software de sistema? Ou, ainda, o que diferencia software de programa?



3

- Software, programas e sistemas
- Algoritmos e linguagens
- O jargão na área de TI
- Ciclo de vida do desenvolvimento de software (metodologias)

4

Software, programas e sistemas

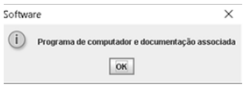
5

Software e programas

- Software consiste em:
 - (1) instruções (programas de computador) que, quando executadas, fornecem características, funções e desempenho desejados
 - (2) estruturas de dados que possibilitam aos programas manipular informações adequadamente
 - (3) informação descritiva, tanto na forma impressa quanto na virtual, descrevendo a operação e o uso dos programas

6

- Software
 - Programa
 - Sequência de passos ou instruções descritos por um algoritmo
 - Algoritmo
 - Estrutura passo a passo de como um problema deve ser resolvido
 - Ex: uma receita de bolo




7

- Software de Sistema (Básico)
 - Sistemas Operacionais
 - Sistemas tradutores
- Software Utilitário
 - Software Aplicativo

8

Software de Sistema (Básico)


- Coleção de programas para apoiar outros programas
- Sistema operacional
- Softwares utilitários ou ferramentas de sistema



9

Software utilitário


- Fornecem ao usuário ferramentas para organizar os discos, verificar disponibilidade de memória, corrigir falhas de processamento
- Úteis ao sistema computacional
- Ex: antivírus, compactadores, emuladores, desfragmentadores, formatadores, backup



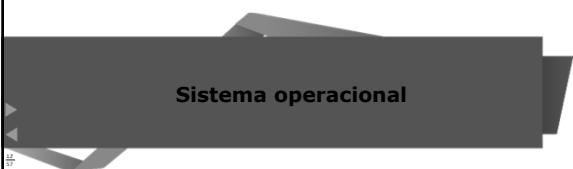
10

Software aplicativo

- Programas que auxiliam o usuário de computador a realizar determinada tarefa
- Popularmente conhecidos como *app*
- Ex: navegadores, editor de texto, planilhas, apresentações, SGBDs, editor de imagens etc.



11



Sistema operacional

12

Sistema operacional

- **Sistema:** conjunto de partes que se interagem para alcançar determinado objetivo
- O que diferencia software de sistema é que um sistema pode ser definido como um conjunto de softwares que interagem entre si para alcançar determinado objetivo

Sistemas operacionais

- O software básico é aquele necessário para o funcionamento do hardware ou de parte dele
- Ex: sistemas operacionais e sistemas tradutores



Sistemas operacionais

- Software responsável pela supervisão dos processos executados em um computador
- Gerencia todo o hardware e todo o software do computador e realiza a "comunicação" entre eles



Sistemas operacionais

Processo: programa em execução

Abrir o gerenciador de tarefas: **CTRL + SHIFT + ESC**
CTRL+ALT+DEL >>
Gerenciador de Tarefas
Pesquisa do Windows >>
Gerenciador de Tarefas



Sistemas operacionais

- Camada de software entre o hardware e os softwares usados pelos usuários



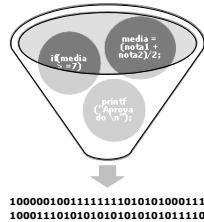
Funções do SO

- Facilitar o uso do computador pelo usuário, tornando mais simples a utilização de seus recursos
- Gerenciar os recursos do computador
- Controlar a execução de programas pela CPU



Sistemas tradutores

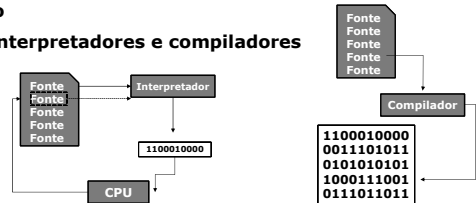
- Converte os programas escritos para um código em uma linguagem de máquina, mais adequada para manipular bits



19

- Programa escrito em linguagem de alto nível: necessidade ser traduzido para a linguagem de máquina para que o computador possa executá-lo

- Interpretores e compiladores



20

Algoritmos e Linguagens

21

Algoritmos

- "Algoritmo é uma sequência de passos que visa atingir um objetivo bem definido." (Forbellone, 1999)

22

- Objetivo: representar mais fielmente o raciocínio envolvido na lógica de programação
- Uma vez concebida uma solução, esta pode ser traduzida para qualquer linguagem de programação

23

Sintaxe x semântica

- Gap semântico entre lógica de programação e a lógica do dia a dia
- Semântica na linguagem de programação: referindo ao conteúdo, ao significado
- Sintaxe está relacionada com as regras, premissas, restrições
- Modo como as palavras podem ser combinadas e formar os enunciados define a sintaxe

24

Sintaxe

- Exemplo 1:
 - ✓ Fiz prova de FDS na semana passada
 - Se escrevêssemos:
 - ✓ FDS passada semana prova fiz
- Exemplo 2:
 - ✓ (2] - /

Semântica

- Define o significado formal das expressões, comandos ou unidades de programas
- Erros de semântica estão relacionados a lógica de programação
- Exemplo: expressão sintaticamente correta na linguagem Python, mas semanticamente não faz sentido somar um número e um caractere
- `a = 2 + "3"`

25

26

Algoritmos – exemplo 1

- Receita de bolo de caneca
- Ingredientes
 - ✓ 2 colheres (sopa) de achocolatado
 - ✓ 3 colheres (sopa) de farinha de trigo
 - ✓ 3 colheres (sopa) de açúcar
 - ✓ meia colher (chá) de fermento em pó
 - ✓ 3 colheres (sopa) de Leite
 - ✓ 2 colheres (sopa) de óleo
 - ✓ 1 ovo (...)



Africa Studio/Shutterstock,
Townday/Shutterstock

(...)

- Modo de preparo
 - Em uma caneca com capacidade superior a 350 ml, misture o achocolatado, a farinha de trigo, o açúcar e o fermento em pó
 - Acrescente o Leite, o óleo e o ovo e misture delicadamente até incorporar
 - Coloque a caneca sobre um prato de sobremesa e leve ao forno micro-ondas por 3 minutos em potência alta
 - Sirva a seguir



Africa Studio/Shutterstock,
Townday/Shutterstock

27

28

Algoritmos – exemplo 2

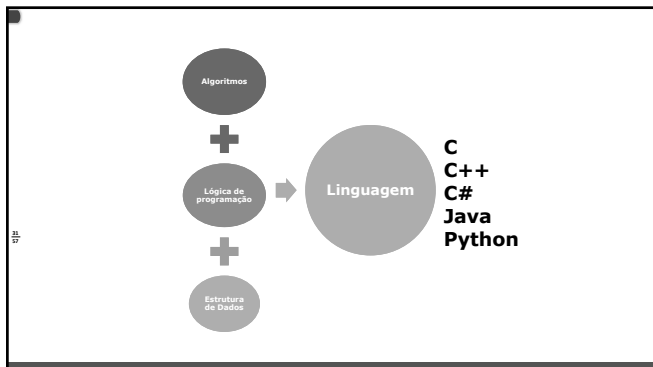
- Calcular média de prova
 - Obter as duas notas de provas
 - Calcular a média aritmética
 - Se a média for igual ou maior que 7, o aluno foi aprovado
 - Se não, ele foi reprovado

Linguagens

- Linguagens de programação tratam os dados de um computador por meio do uso de algoritmos
- Programador:
 - Deve encontrar um algoritmo que resolve o seu problema
 - Implementá-lo usando uma linguagem de programação

29

30



31

Exemplo - Python

```

#Obter as duas notas de provas
nota1 = float(input("Digite a primeira nota: "))
nota2 = float(input("Digite a segunda nota: "))
#Calcular a média aritmética
media = (nota1 + nota2)/2
#Se a média for igual ou maior que 7, o aluno foi aprovado
if media >= 7:
    print("Aprovado")
#Se não, ele foi reprovado
else:
    print("Reprovado")
  
```

32

Exemplo - Java

```

//Obter as duas notas de provas
System.out.println("Digite a primeira nota: ");
float nota1 = teclado.nextFloat();
System.out.println("Digite a segunda nota: ");
float nota2 = teclado.nextFloat();
//Calcular a média aritmética
float media = (nota1 + nota2)/2;
//Se a média for igual ou maior que 7, o aluno foi aprovado.
if (media >= 7)
    System.out.println("Aprovado");
//Se não, ele foi reprovado
else
    System.out.println("Reprovado");
  
```

33

Exemplo - C

```

//Obter as duas notas de provas
printf("Digite a primeira nota: ");
scanf_s("%d", &nota1);
printf("Digite a segunda nota: ");
scanf_s("%d", &nota2);
//Calcular a média aritmética
media = (nota1 + nota2) / 2;
//Se a média for igual ou maior que 7, o aluno foi aprovado
if (media >= 7)
    printf("Aprovado");
//Se não, ele foi reprovado
else
    printf("Reprovado");
  
```

34

Glossário do desenvolvedor de software

35

Glossário do desenvolvedor de software

- Significado de Jargão
- "Linguagem restrita a determinado grupo profissional ou social; gíria. Linguagem de códigos que alguns grupos usam para que pessoas desconhecidas não compreendam suas conversas; gíria."
- JARGÃO. In: DÍCIO, Dicionário Online de Português. Porto: 7Graus, 2022. Disponível em: <<https://www.dicio.com.br/jargao/>>. Acesso em: 6 dez. 2022

36

Glossário do desenvolvedor de software

- **Front-end:** parte gráfica de uma aplicação web
- **Back-end:** desenvolvimento no lado do servidor
- **Full-stack:** ambas as abordagens

37

- **API (Application Programming Interface)**
 - Conjunto de rotinas e padrões de programação
 - Objetivo: acessar aplicativos de software
 - Plataformas baseados na web
 - Utilizada por programa/aplicação
- **Framework:**
 - Conjunto de código de LP específica
 - Auxilia no desenvolvimento web ou de software
 - Biblioteca de códigos com funções já prontas
 - Arcabouço de código



38

- **IDE (Integrated Development Environment):**
 - Integra diversas funcionalidades para desenvolvimento em única interface gráfica
 - Auxilia e agiliza o processo de desenvolvimento
- **SDK (Software Development Kit):**
 - Composição: compilador, debugger e API
 - Conjunto de ferramentas fornecidos por um fabricante para que se desenvolva para uma plataforma ou sistema específico



39

- **Nativo**
 - Desenvolvido para uma única plataforma
 - Utilização de linguagens e ferramentas específicas para a plataforma em questão
 - Exemplo: para Android, usa-se Android SDK; e Android Studio, linguagem Java e kotlin
- **Híbrido**
 - Implementação utiliza html, css e Javascript
 - Frameworks ou ferramentas que permitem uma mesma base de código
 - Uma linguagem e distribuída para várias plataformas



40

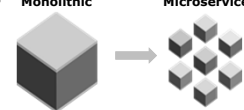
- **Serviços:**
 - processos de software

O termo *processo de software* nesse caso não é no sentido de conjunto de atividades da engenharia de software, mas, sim, naquele programa processado pelo SO

Nome	Estado	11%	85%	8%	8%
Aplicativo (1)		11%	85%	8%	8%
Processo (1)		11%	85%	8%	8%
Processo (2)		11%	85%	8%	8%
Processo (3)		11%	85%	8%	8%
Processo (4)		11%	85%	8%	8%
Processo (5)		11%	85%	8%	8%
Processo (6)		11%	85%	8%	8%
Processo (7)		11%	85%	8%	8%
Processo (8)		11%	85%	8%	8%
Processo (9)		11%	85%	8%	8%
Processo (10)		11%	85%	8%	8%
Processo (11)		11%	85%	8%	8%
Processo (12)		11%	85%	8%	8%
Processo (13)		11%	85%	8%	8%
Processo (14)		11%	85%	8%	8%
Processo (15)		11%	85%	8%	8%
Processo (16)		11%	85%	8%	8%
Processo (17)		11%	85%	8%	8%
Processo (18)		11%	85%	8%	8%
Processo (19)		11%	85%	8%	8%
Processo (20)		11%	85%	8%	8%

41

- **Monolítico:** roda com um único processo
- **Microserviços:** abordagem arquitetônica e organizacional do desenvolvimento de software na qual o software consiste em pequenos serviços independentes que se comunicam usando APIs bem definidas. São autônomos e especializados



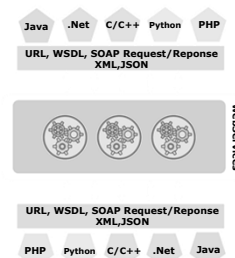
Arquitetura/Shutterstock

42

- **SOAP – Service-Oriented Architecture (Or Application) Protocol**
 - Utiliza arquivos xml
 - Protocolo de transporte: HTTP com
- **REST – Representational State Transfer**
 - Conjunto de restrições para criação de webservices
 - Quando um serviço implementa esse padrão: Restfull
 - Restfull utiliza arquivos JSON

43

- **SOAP e REST são APIs e são arquiteturas de webservice**
- **Webservices são independentes de linguagem**



44

- **Commit:**
 - Enviar alterações de determinado trecho do código
 - Enviar criação de uma nova versão do projeto
- **Versionamento:**
 - Atribuição de número de versão ao estado do projeto
 - Exemplo: pastas com versões do projeto: trabalho_FDS_v1, trabalho_FDS_v2

45

- **Snapshot: cópia instantânea em determinado tempo de um volume**
- **Debug:**
 - Debugging ou debugar
 - Depurar o programa
 - Encontrar erros no programa e tentar resolvê-los



46

- **Git:**
 - Sistema de controle de versão
 - Gerencia as várias versões no desenvolvimento de um documento
 - O logo do GIT representa a ramificação para desenvolvimento não linear
- **Github:**
 - Plataforma de desenvolvedor completa para criar, dimensionar e fornecer software seguro
 - Utiliza o Git como sistema de controle

47

Ciclo de vida de software

48

Ciclo de vida de desenvolvimento de softwares

- Software Development Life Cycle – SDLC
- Modelo de processo
- Representação simplificada de um processo de software

- Quatro atividades são comuns a todos os processos de software: especificação, desenvolvimento, validação e evolução



Sommerville (2019)

Analogia com a construção de casa

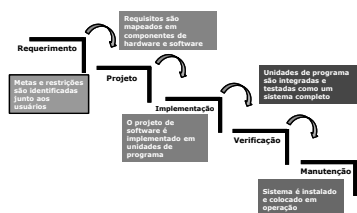


Ciclo de vida de um software



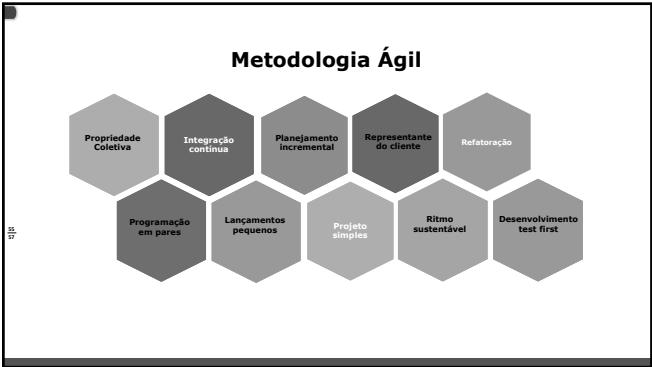
Modelo cascata

- Clássico
- Tradicional
- SDLC
- Waterfall

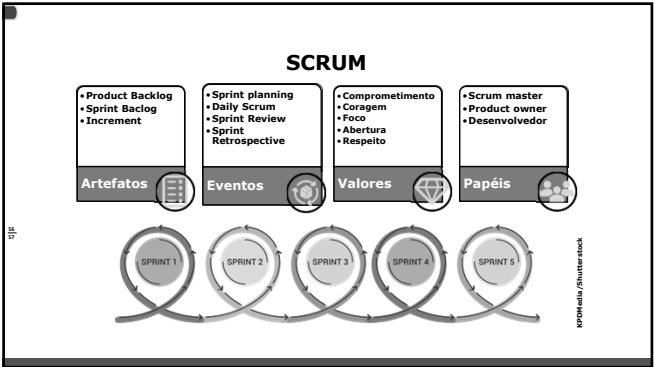


Metodologia Ágil

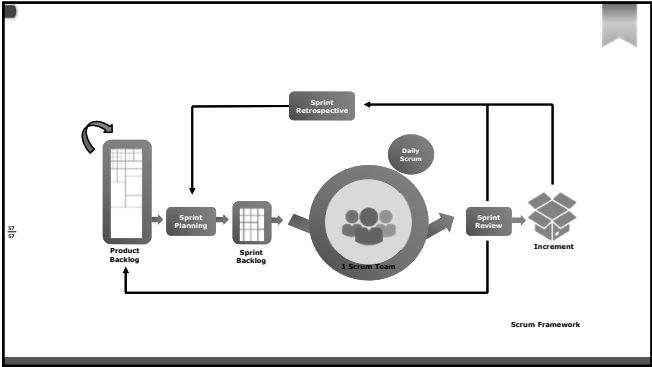
- Alternativa para gestão de projetos tradicionais
- Entrega rápida das funcionalidades
- Foco no software e não no projeto
- Adequada a aplicações em que os requisitos mudam rapidamente
- Duas abordagens: o extreme Programming (XP) e o SCRUM



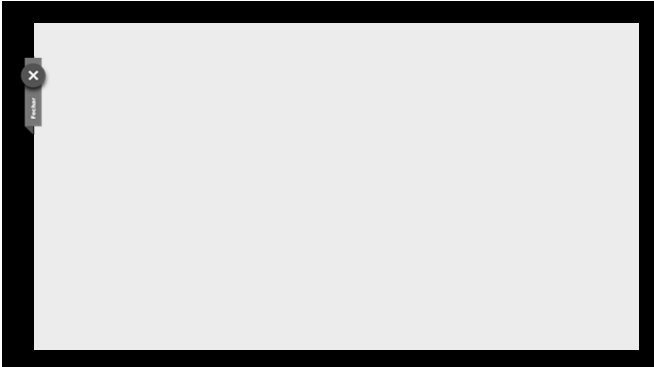
55



56



57



58