

Fundamentos de Lógica e Computadores

O que é um computador

Computador é uma máquina que processa informações (na verdade, *dados*) de forma eletrônica, por meio de cálculos matemáticos complexos (realiza "computação").



Exemplos de Computadores

- **PC (Personal Computer)**
- **iMac**
- **Smartphone**
- **Mainframe**
- **Servidor**
- **Notebook**
- **Tablet**
- **Supercomputador**

Arquitetura de um computador

Um computador (PC, Servidor, Notebook, Smartphone, etc.) é composto de duas "partes" principais:

- Hardware
- Software



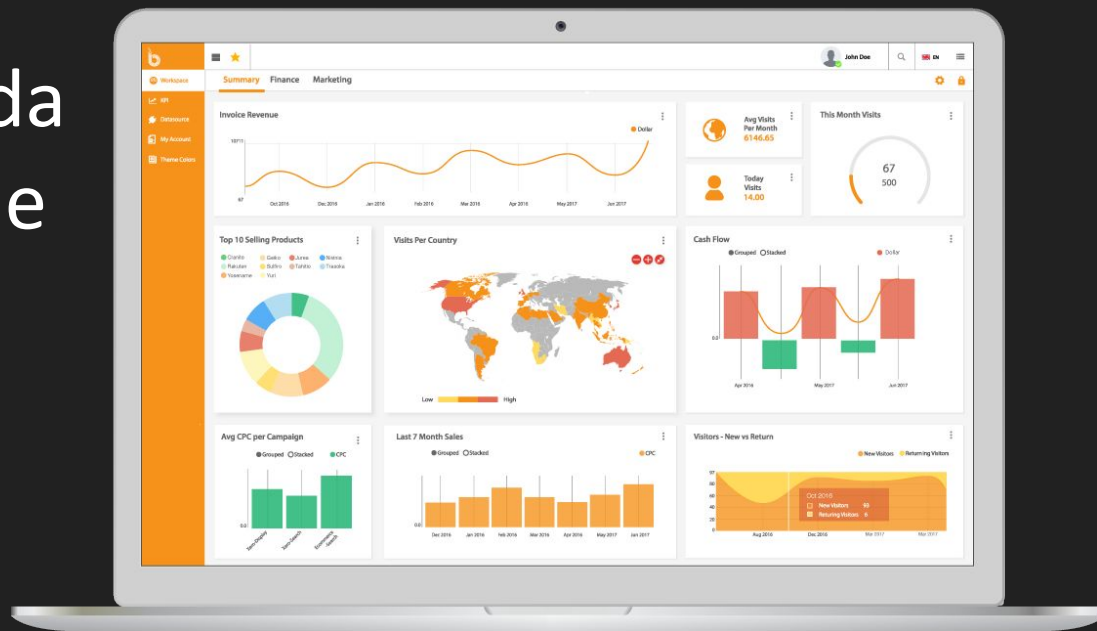
Hardware



- Parte física do sistema, composta pelos componentes eletro-eletrônicos e dispositivos como monitor, teclado, CPU, memória RAM, discos, SSD, etc.

Software

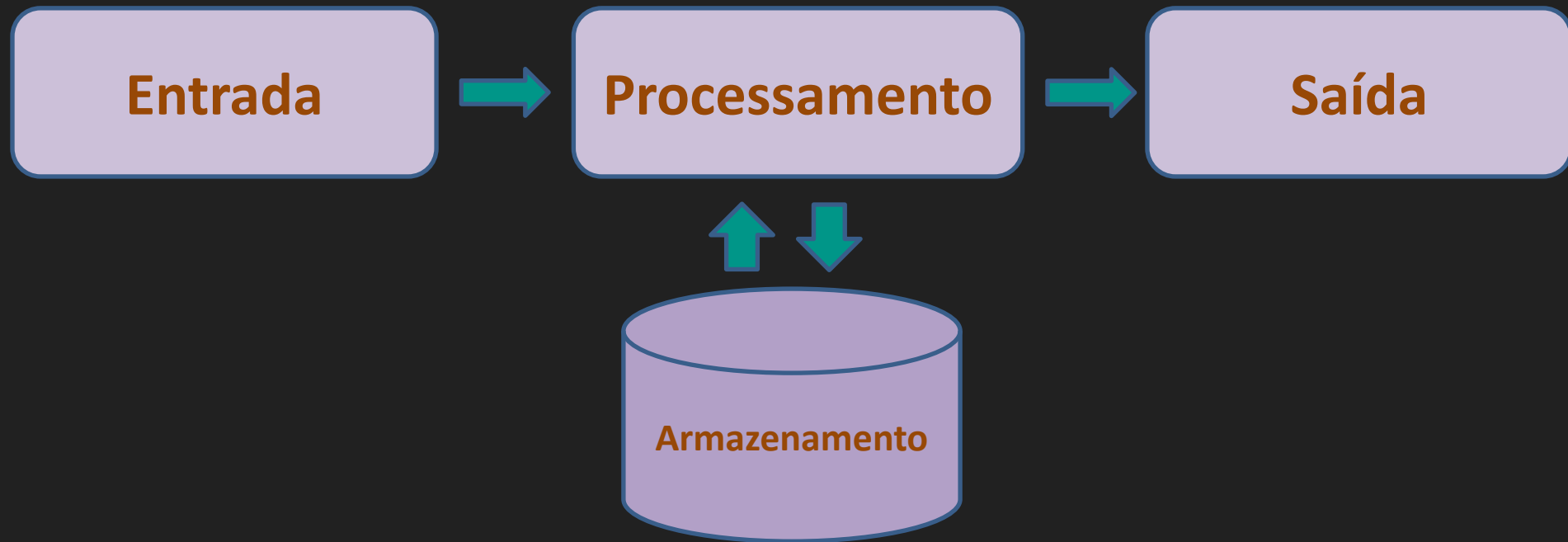
- Parte lógica do sistema representada pelos programas que rodam sobre o hardware da máquina.
- Inclui o S.O.



Exemplos: Hardware e Software

Hardware	Placa-mãe Processador Memória RAM Teclado Chipset	Mouse Monitor Placa de Rede Fonte Webcam
Software	Sistema Operacional Editor de Texto Planilha Eletrônica Navegador GUI	Jogos Antivírus Apps de Celular IDE Editor de Imagens

Processamento de Dados



O que é Lógica de Programação

O que é Lógica de Programação?



O que é um Programa?

- Conjunto de instruções executadas em uma ordem conveniente que contém as tarefas ou operações necessárias para resolver um problema, usando um computador ou equipamento similar.

Os programas são executados por uma unidade central de processamento (CPU) na memória (RAM) do computador.



O que é Programação?

- Programação de computadores é o processo de realização de tarefas específicas para resolução de problemas por meio da construção de um programa de computador.
- A programação se dá pela escrita de códigos estruturados com o emprego de ferramentas de software e linguagens específicas, as linguagens de programação.
- Exemplos de linguagens usadas em programação incluem: Python, C++, JavaScript, Rust, PHP e Java.

O que é Lógica?

- Lógica, de acordo com o dicionário *Oxford Languages*, é ***um raciocínio conduzido ou avaliado de acordo com princípios estritos de validade.***
- E também o ***estudo das leis do pensamento ou raciocínio correto.***
- E ainda, um ***conjunto de regras e técnicas para distinguir o raciocínio correto do incorreto.***

O que é Lógica de Programação?



Lógica de Programação é um Paradigma de *Programação* baseado em *Lógica*, que emprega sentenças estruturadas para construção de algoritmos que realizam tarefas específicas

Sistema ou conjunto de princípios que descreve como os elementos devem ser organizados para que um computador possa executar tarefas específicas.

Linguagens de Programação

- Uma Linguagem de Programação é um conjunto de regras utilizadas para criar programas de computador.
- Essas regras são usadas para expressar os algoritmos de forma precisa.
- Os algoritmos são escritos em uma linguagem de programação, gerando um código-fonte.

Há três níveis principais de linguagens de programação.

Linguagens de Programação: Níveis

- **Linguagem de Máquina:** linguagem compreendida pelo computador. As instruções são representadas por valores binários (bits), 0 e 1
- **Linguagem de Baixo Nível:** linguagem que utiliza instruções próximas à linguagem de máquina, porém compreensíveis por seres humanos. Ex.: Assembly.
- **Linguagem de Alto Nível:** linguagem cujas instruções estão bem próximas da compreensão humana. Ex: C#, Python, Visual Basic, Java, LISP, PHP, C, C++

Do Código ao Programa

- Uma linguagem de programação, para ser efetivamente transformada em um programa e executada, pode ser convertida em código de máquina executável ("**compilação**") ou interpretada ("**interpretação**")



Compilação

- Método que traduz todo o código-fonte em um ou mais arquivos que podem ser armazenados e executados quantas vezes se desejar, sem necessitar de nova compilação.
- Usa um Compilador (software específico)
- Exemplo: Linguagens C/C++.

Interpretação

- Os comandos necessários contidos no código-fonte vão sendo executados à medida em que são traduzidos, sendo necessária a interpretação toda vez que se quiser rodar o programa.
- Usa um interpretador (software)
- Exemplos: Python, C#, Java

O que é um Algoritmo

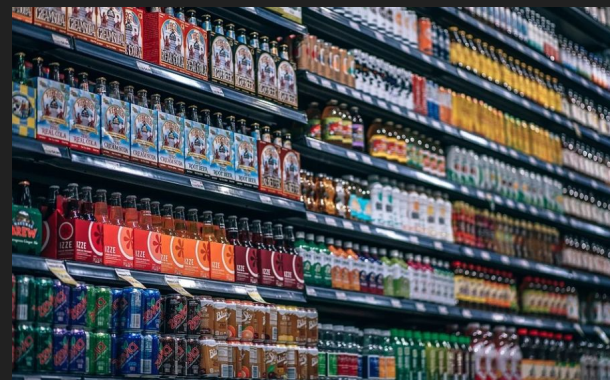
Sequência Lógica

- Passos ou tarefas executadas até atingir um objetivo ou solucionar um problema apresentado.



Sequência lógica: comprar um refrigerante no mercado

1. Entrar no mercado
2. Pegar um carrinho (ou cestinha)
3. Andar até a seção de refrigerantes
4. Pegar uma garrafa de refrigerante na gôndola
5. Colocar a garrafa no carrinho
6. Dirigir-se à fila do caixa
7. Pagar pelo refrigerante
8. Sair do mercado



Verificar essa sequência: está certa?

1. Pegar um carrinho
2. Dirigir-se à fila do caixa
3. Sair do mercado
4. Pegar uma garrafa de refrigerante na gôndola
5. Entrar no mercado
6. Colocá-la no carrinho
7. Pagar pelo refrigerante
8. Ir até a seção de refrigerantes



Algoritmo

- Sequência de instruções **ordenada**, **finita** e **não ambígua** que são executadas por um sistema para realizar uma tarefa.
- Um algoritmo deve ter fim, poder se comunicar com o mundo exterior, e executar suas instruções em um tempo finito.
- Cada linha de um algoritmo deve executar uma única instrução (tarefa / ação).
- Como uma "receita de bolo"

[illegible]

Algoritmo: Exercício

- Montar um algoritmo para preparar um café utilizando papel de filtro (coador) e uma caneca no fogão.



Resolução sugerida

1. Dirija-se ao armário
2. Abra o armário
3. Pegue um coador
4. Pegue o papel de filtro
5. Coloque o papel de filtro no coador
6. Pegue uma garrafa térmica
7. Coloque o conjunto sobre a garrafa térmica
8. Pegue o pó de café no armário
9. Coloque 5 colheres de pó no coador
10. Pegue uma caneca
11. Encha a caneca de água
12. Coloque a caneca sobre uma boca do fogão
13. Acenda a boca do fogão
14. Espere a água ferver
15. Retire a caneca do fogão
16. Apague o fogão
17. Despeje a água da caneca no coador
18. Espere a água toda escorrer
19. Retire o coador
20. O café está pronto.

Refinamento de ações

- O algoritmo para fazer o café é funcional, porém há muitas outras ações que devem ser realizadas, e outras tantas que podem ser desdobradas em ações menores. Essas ações são chamadas de *não-primitivas*.
- Já ações que não podem mais ser desdobradas são chamadas de *primitivas*.

Exemplo de refinamento

Ação: Ferver a água:

- Refinamento:
 - Pegar a caneca
 - Levar à pia
 - Colocar sob a torneira
 - Abrir a torneira
 - Encher a caneca de água
 - Fechar a torneira
 - Levar a caneca ao fogão
 - Acender o fogão
 - Esperar a água ferver



Exercício: Refinamento

Refine a ação a seguir:

Acender o fogão



Resolução: Refinamento

Ação: *Acender o fogão*

- Refinamento:
 - Verificar se gás está ligado
 - Se gás não estiver ligado, ligá-lo
 - Localizar a boca a ser usada
 - Girar o botão correspondente para abrir o gás
 - Pressionar o botão para acender a boca
 - Enquanto fogo não acender, continuar pressionando botão
 - Após fogo aceso, soltar botão
 - Colocar a caneca sobre a boca acesa para fervura

Representação de Algoritmos

Como representar um algoritmo

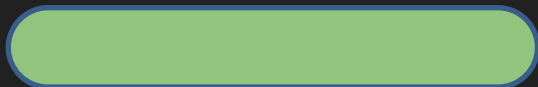
Há diversas maneiras de se representar um algoritmo:

- Fluxograma
- Diagrama de Chapin
- Português Estruturado
- Usando Linguagens de Programação

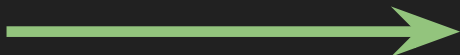
Fluxograma

- Diagrama que descreve o fluxo de dados e instruções usando desenhos geométricos básicos que simbolizam a entrada, saída e processamento de dados.
- Indicado para algoritmos de pouca extensão, porém complexos.

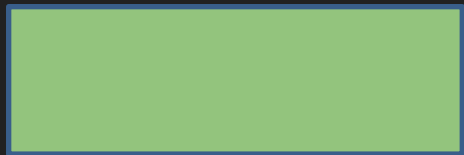
Símbolos básicos para fluxogramas



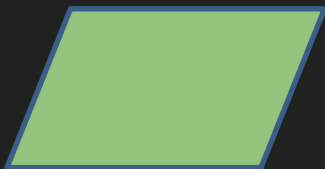
Início ou fim de programa ou subprocesso



Seta de fluxo de dados (ordem de execução)

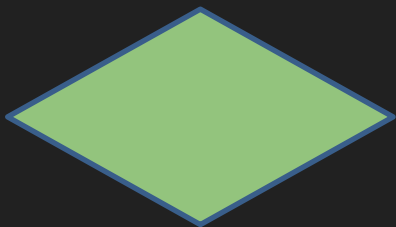


Bloco de processamento e manipulação de dados



Entrada e saída de dados

Símbolos básicos para fluxogramas



Estrutura de decisão



Conector de partes do diagrama (on-page)



Conector de página (off-page)

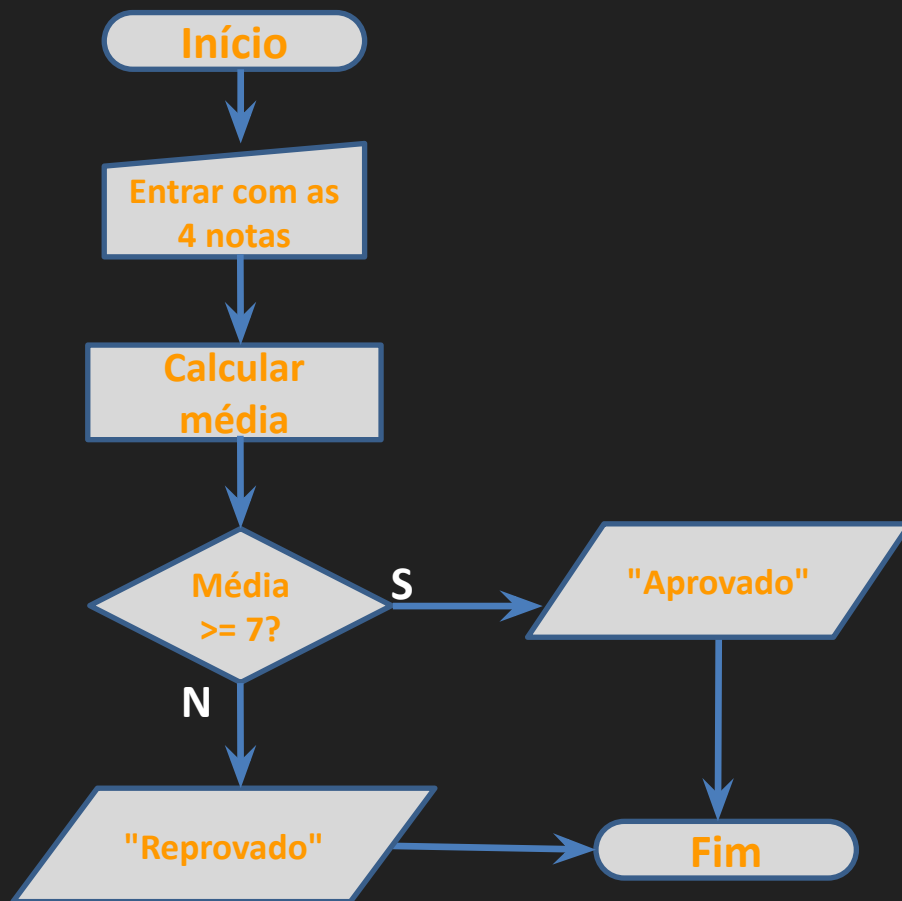


Entrada manual de dados (via teclado)

Exemplo de Fluxograma

- Problema: calcular a nota média de um aluno, baseado nas notas de quatro provas, e verificar se foi aprovado.
- Critério: será aprovado se a nota média for igual ou superior a 7.0

Resolução: Fluxograma



Resolução: Sequência Lógica (Algoritmo)

início

Entrar com as 4 notas

Calcular média aritmética das notas

Verificar se a média é maior ou igual a 7.0

Se sim, mostrar mensagem "Aprovado"

Se não, mostrar mensagem "Reprovado"

fim

Diagrama de Chapin

- Desenvolvido por Isaac Nassi e Ben Shneiderman e ampliado por Ned Chapin
- Substitui o fluxograma tradicional por um diagrama de quadros que apresenta uma visão hierárquica e estruturada da lógica do programa.
- Facilita a codificação do algoritmo bem como a representação de certas técnicas, como a recursividade.
- Também conhecido como "**Estrutograma**".

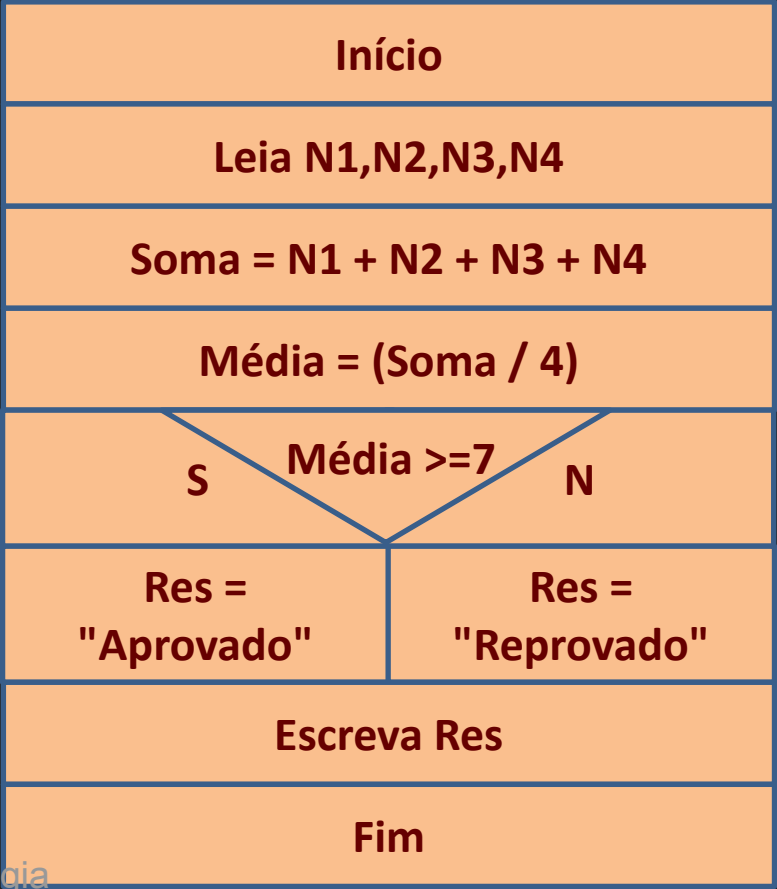
Resolução: 3. Diagrama de Chapin



Ben Shneiderman



Isaac Nassi



Ned Chapin

Português Estruturado

- Técnica narrativa baseada em **PDL** – *Program Design Language* (Linguagem de Projeto de Programa).
- Notação para elaboração de algoritmos estruturados que serão utilizados para a criação e desenvolvimento em uma linguagem de programação qualquer.
- Também conhecido como ***Pseudocódigo***

Resolução: Pseudocódigo

```
programa {  
    cadeia res  
    real N1, N2, N3, N4, media  
  
    funcao inicio()  
    {  
        leia(N1, N2, N3, N4)  
        media = (N1 + N2 + N3 + N4) / 4  
  
        se (media >= 7) {  
            res = "Aprovado"  
        }  
        senao {  
            res = "Reprovado"  
        }  
  
        escreva("Resultado: " + res)  
    }  
}
```

Instalação do Portugol Studio

Instalação do Portugol Studio

Instalação no Windows e Linux

- Download: <http://lite.acad.univali.br/portugol/>

Alternativa para estudo: Portugol Web Studio:

<https://portugol-webstudio.cubos.io/ide>

<https://dgadelha.github.io/tcc-webstudio/> (nova versão experimental)