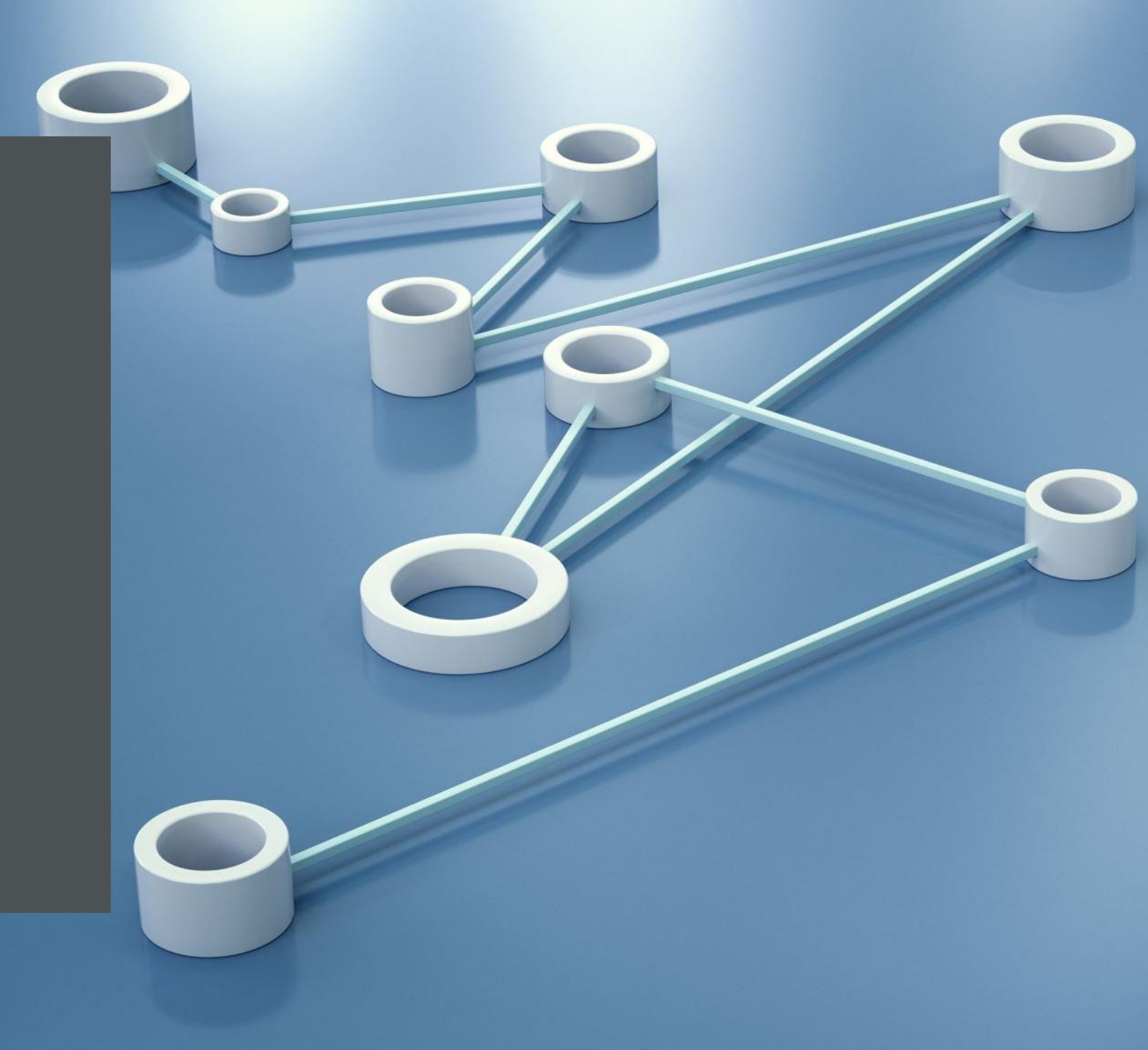




Tipo Abstrato de Dado

Taniro Rodrigues

tanirocr@gmail.com

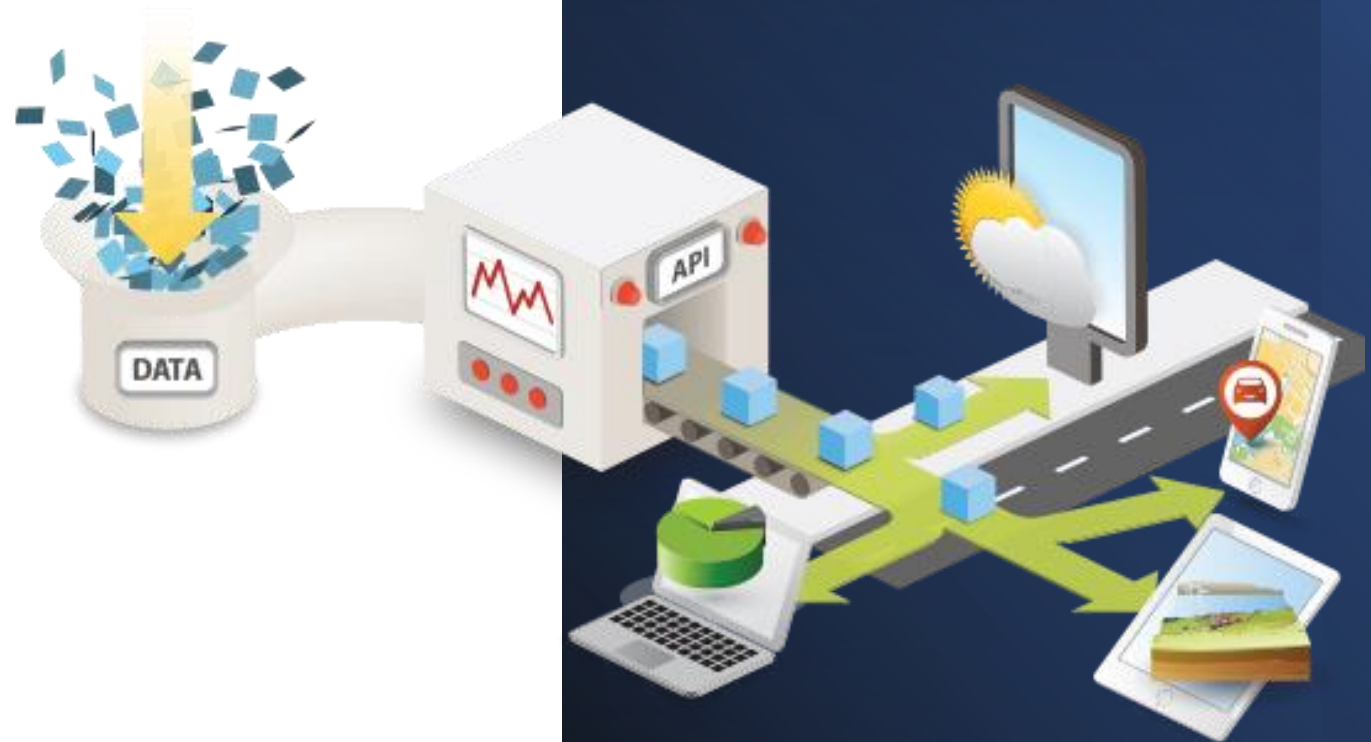


O que é Informação?

O conceito de **informação** pode ser genericamente ligado as **noções de dados, instruções, conhecimento, significado**, entre outros.

Informação

- O computador é uma máquina que **manipula informação**.
- A informática é a ciência que estuda o **tratamento da informação** através do uso de dispositivos de **processamento de dados**.

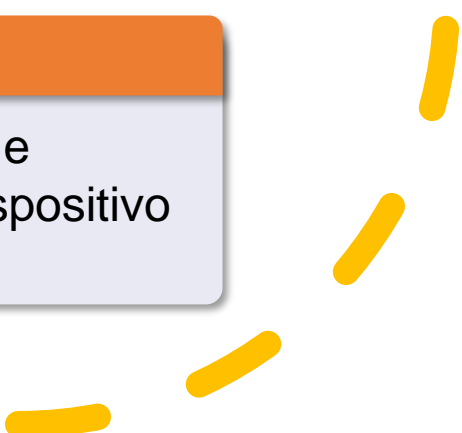


Representação da Informação

- É importante entender os conceitos de organização e gerenciamento de informações **dentro de um dispositivo de processamento de dados.**
- Como representar dados?

Estruturas de dados

Formas armazenamento, organização e tratamento das informações em um dispositivo de processamento de dados.



Representação da Informação

- Assim como no mundo real, é possível ter diversas **representações diferentes** para uma mesma informação.
- Uma delas poderá lidar com um problema de **melhor forma de a outra**.

Dados

As linguagens de programação **adaptam, ou abstraem**, os conceitos abstratos do nosso dia-a-dia em conceitos que podem ser armazenados nos computadores. Exemplos: números, letras, frases, etc.



nina edelweiss
renata galante



Representação da Informação

*“Um fator que determina o papel dessas estruturas no processo de programação de aplicações é a **identificação do quão bem as estruturas de dados coincidem com o domínio do problema a ser tratado.**”*

Estruturas de Dados - Série Livros Didáticos Informática UFRGS

Edelweiss, Nina; Galante, Renata

Linguagens de Programação

- Necessidade de suporte a uma variedade de **tipos** e **estruturas** para **representar os dados** manipulados pelas aplicações.
- Um tipo de dado é dado com base num conjunto de valores que uma variável pode assumir.
- Cada nome (identificador) em um programa, em qualquer linguagem, tem um tipo associado a ele.
- **Esse tipo determina as operações que podem ser aplicadas ao como e como elas devem ser interpretadas.**

1 + 1

'1' + '1'

Tipos de Dados



- Primitivos/básicos
- Estruturados homogêneos
- Estruturados heterogêneos

Dados Primitivos



- Também chamados de estruturas de dados primitivas ou **atômicos**.
- São aqueles que estão **disponíveis na maioria dos computadores**.
 - Inteiros, Reais, Lógicos, Caracteres

Inteiros

- **Conjunto de valores:** negativos, zero, positivos;
- **Operações:** +, -, *, /;
- **Comparações:** ==, >, <, ==>, <=, !=

Reais

- **Conjunto de valores:** parte inteira e fracionária;
- **Operações:** +, -, *, /;
- **Comparações:** ==, >, <, ==>, <=, !=

Lógicos


- **Conjunto de valores:** verdadeiro e falso;
- **Operações:** e, ou e não
- **Comparações:** ==, !=

Dados Estruturados Homogêneos

- São aqueles que permitem agregar mais do que o valor de uma variável, existindo uma relação estrutural entre os seus elementos.
- Exemplos:
 - Arrays, Matrizes, Cadeira de caracteres, registros, etc.

Arranjo

- Um Arranjo é uma estrutura de **dados linear** que abriga um número definido de dados do mesmo tipo e que necessita da **especificação de um índice** para que seus elementos possam ser acessados.
- Cada uma das células de um vetor **possui seu próprio endereço de memória**.
- Dados são **alocados** de maneira **sequencial**.



peçoas	5	7	0	5	6	0	1	8
	0	1	2	3	4	5	6	7

Arranjo

- A partir do endereço do **primeiro elemento** é possível determinar a localização dos demais elementos do vetor.
- **Lado a lado da memória e tamanho fixo.**

Posição N = endereço_inicial + N * tamanho_dado

idades	5	7	0	5	6	0	1	8
	0	1	2	3	4	5	6	7

Matriz

- Vetor com **duas ou mais dimensões** de alocação **estática e sequencial**.
- Necessita de, pelo menos, um **índice** para cada dimensão para acesso aos dados.
- Como o vetor, **possui tamanho fixo e todos os elementos são do mesmo tipo**.

`pessoas[0][4]`

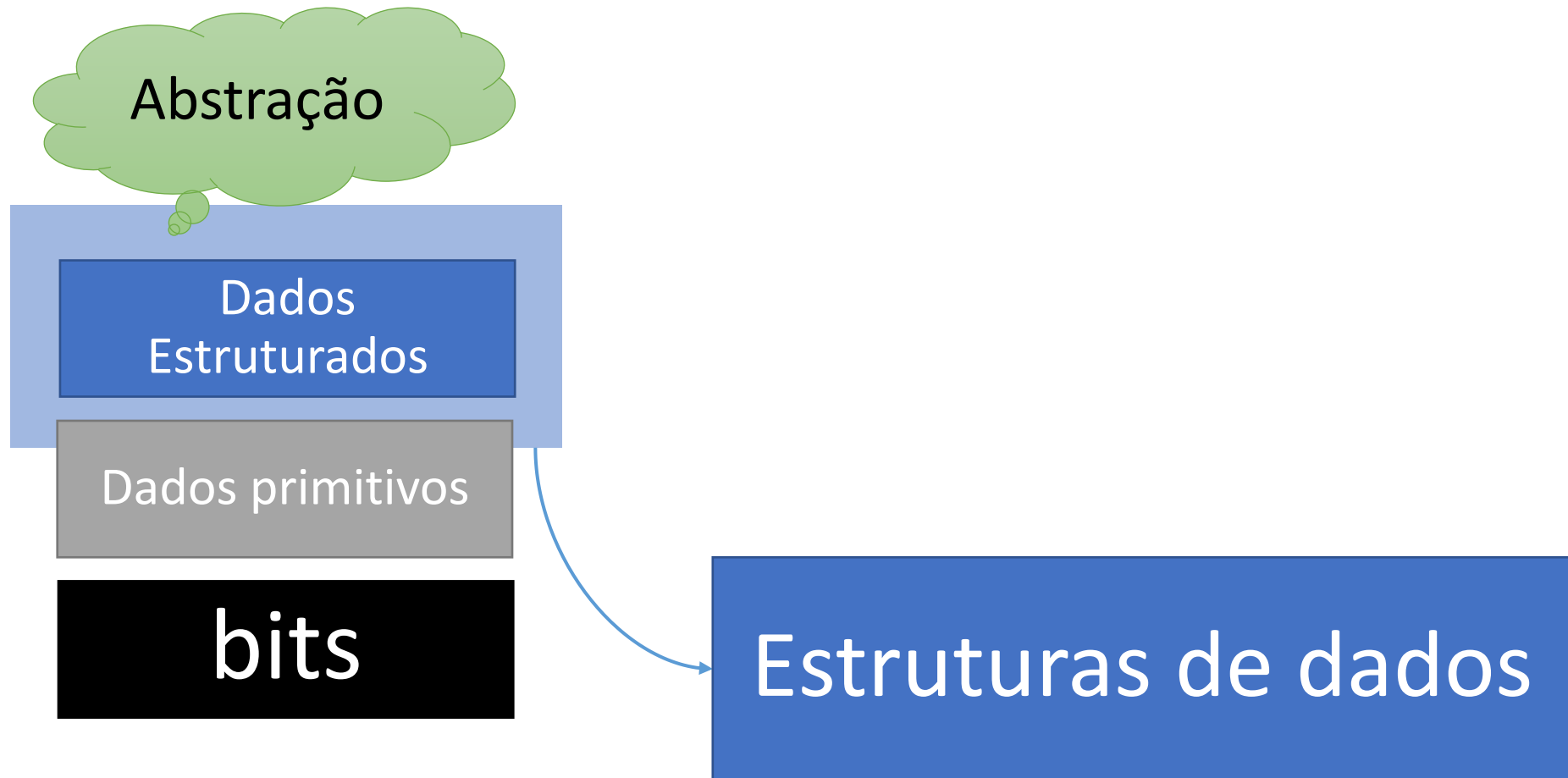


Dados Estruturados Heterogêneos

```
Pessoa {  
    Texto nome;  
    Número idade;  
    Texto endereço;  
}
```

-
- São compostos por certo número de campos de dados que são itens de dados individuais.
 - Visa **facilitar o agrupamento de variáveis** que não são do mesmo tipo, mas que **guardam estreita relação lógica**.
 - Cada tipo de dado armazenado em um registro é chamado de **campo**.

Tipos de Dados



Abstração

“[...] A abstração é nossa mais importante ferramenta mental para lidar com a complexidade. Portanto, um problema complexo não poderia ser visto imediatamente em termos de instruções de computador [...] mas, antes, em termos de entidades naturais ao próprio problema, abstraído de maneira adequada.”

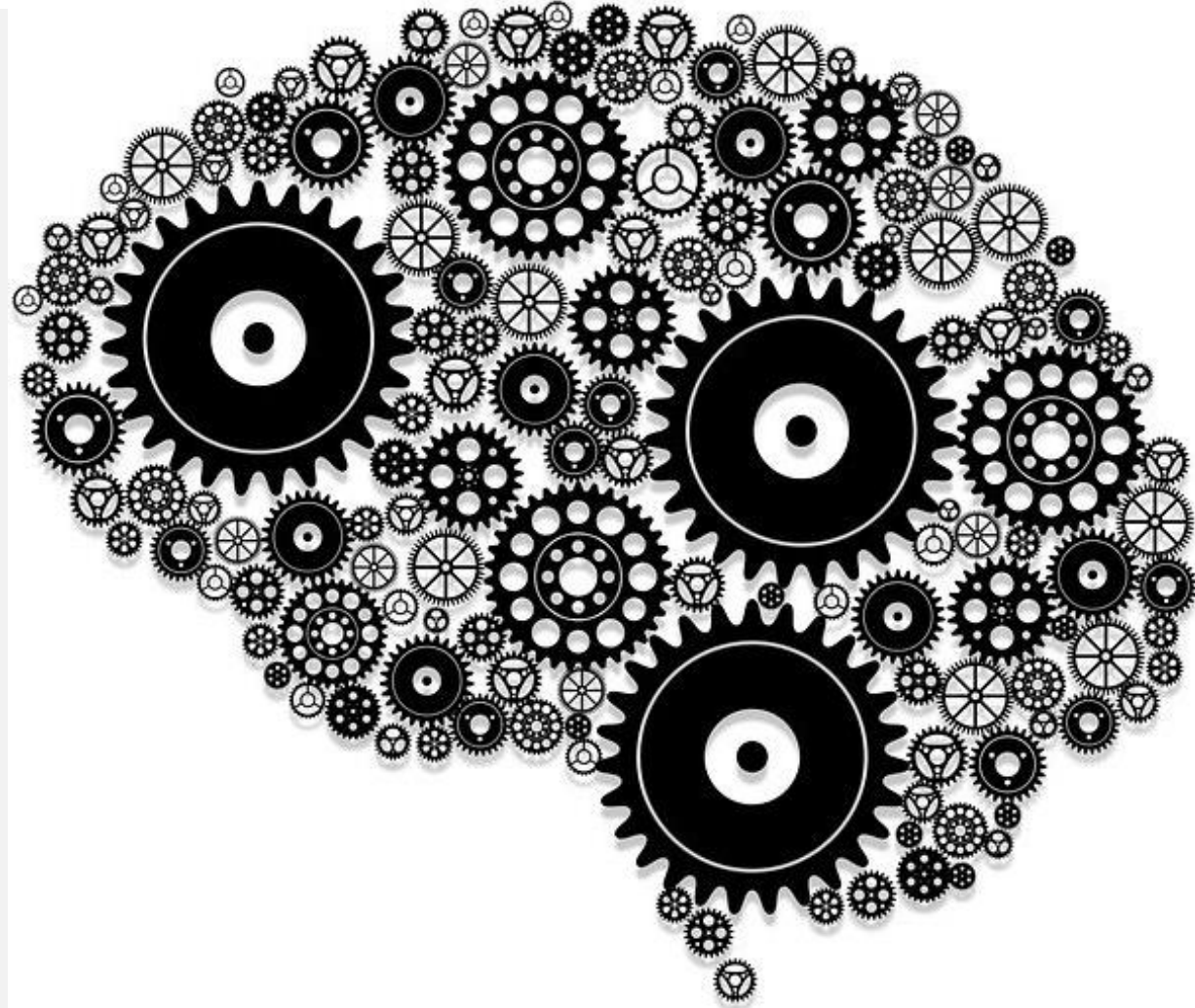


Niklaus Wirth (1989)

Cientista da computação que projetou diversas linguagens de programação.

Tipo Abstrato de Dado

- Abstrair é **separar mentalmente** uma parte de um todo para a considerar independente das outras partes.
- Podemos dizer que o computador é uma máquina abstrata, pois independente de saber **como** ele executa as tarefas, interessa saber **o que** ele pode executar.



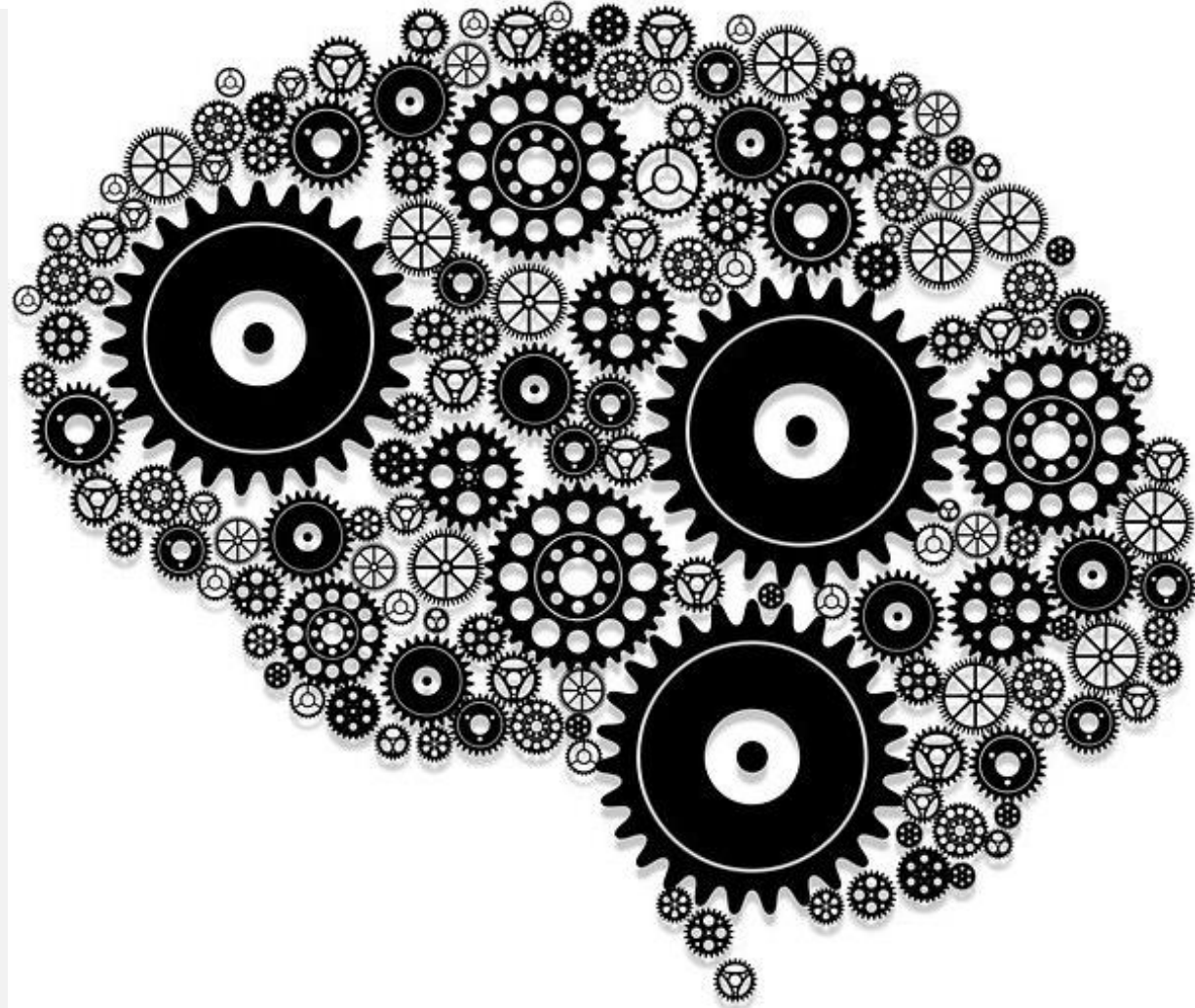
Tipo Abstrato de Dado

- Na **engenharia de software** adota-se muitas vezes a divisão do problema em módulos, estabelecendo os níveis de abstração.

Interface

Controle

Banco de Dados



Tipo Abstrato de Dado

- Dominar a habilidade de **pensar de forma abstrata** sobre os componentes de um programa de computador é um **problema** para programadores iniciantes.
- Enquanto essa **habilidade não é desenvolvida**, o estudante vê o programa como uma **coleção não estruturada de declarações e expressões**.
- Com o desenvolvimento da habilidade de abstrair o programador tende a ver o programa como uma **coleção de funções e classes**.

Tipo Abstrato de Dados

- Os Tipos Abstratos de Dados (ou TADs) são **estruturas de dados** capazes de **representar os tipos de dados** que não foram previstos nas linguagens de programação e que são necessários para representar um **domínio** em específico.
- É um tipo de dado que agrega os seguintes componentes:
 - Definição abstrata dos dados;
 - Ações abstratas
- Para se considerar abstrato o tipo de dados, devemos aplicar as **operações** definidas para ele independentemente de saber **como foram implementadas** ou **como os dados foram armazenados**.

Bibliografia Básica

- CORMEN, Thomas H et al. **Algoritmos: teoria e prática**. Rio de Janeiro: Elsevier, 2012. 926 p. ISBN: 9788535236996.
- ASCENCIO, Ana Fernanda Gomes. **Estruturas de dados: algoritmos, análise da complexidade e implementações em Java e C/C++**. São Paulo: Pearson, c2010. 432 p. ISBN: 9788576052216, 978857605816.
- PIVA JÚNIOR, Dilermando (et al). **Estrutura de dados e técnicas de programação**. 1. ed. Rio de Janeiro, RJ: Campus, 2014. 399 p. ISBN: 9788535274370.

Bibliografia Complementar

- FERRARI, Roberto et al. **Estruturas de dados com jogos**. 1. ed. Rio de Janeiro: Elsevier, 2014. 259p. ISBN: 9788535278040.
- GRONER, Loiane. **Estruturas de dados e algoritmos em Javascript**: aperfeiçoe suas habilidades conhecendo estruturas de dados e algoritmos clássicos em JavaScript. São Paulo: Novatec, 2017. 302 p. ISBN: 9788575225530.
- SZWARCFITER, Jayme Luiz; MARKENZON, Lilian. **Estruturas de dados e seus algoritmos**. 3. ed. Rio de Janeiro: LTC, 2010. xv, 302 p. ISBN: 9788521617501.
- GOODRICH, Michael T; TAMASSIA, Roberto. **Estruturas de dados e algoritmos em Java**. 5. ed. Porto Alegre: Bookman, 2013. xxii, 713 p. ISBN: 9788582600184.
- GUIMARÃES, Ângelo M. **Algoritmos e estruturas de dados**. LTC, 1994.