



Serviço Nacional de Aprendizagem Industrial  
PELO FUTURO DO TRABALHO

SÉRIE TI - SOFTWARE

# LÓGICA DE PROGRAMAÇÃO

## ESTRUTURAS DE DADOS



## **CONFEDERAÇÃO NACIONAL DA INDÚSTRIA – CNI**

*Robson Braga de Andrade*  
Presidente

### **GABINETE DA PRESIDÊNCIA**

*Teodomiro Braga da Silva*  
Chefe do Gabinete - Diretor

### **DIRETORIA DE EDUCAÇÃO E TECNOLOGIA - DIRET**

*Rafael Esmeraldo Lucchesi Ramacciotti*  
Diretor de Educação e Tecnologia

### **SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL - SENAI**

*Robson Braga de Andrade*  
Presidente do Conselho Nacional

#### **SENAI – Departamento Nacional**

*Rafael Esmeraldo Lucchesi Ramacciotti*  
Diretor-Geral

*Julio Sergio de Maya Pedrosa Moreira*  
Diretor-Adjunto

*Gustavo Leal Sales Filho*  
Diretor de Operações



*Serviço Nacional de Aprendizagem Industrial*

**PELO FUTURO DO TRABALHO**

SÉRIE TI - SOFTWARE

# LÓGICA DE PROGRAMAÇÃO

ESTRUTURAS DE DADOS



© 2020. SENAI – Departamento Nacional

© 2020. SENAI – Departamento Regional de Santa Catarina

A reprodução total ou parcial desta publicação por quaisquer meios, seja eletrônico, mecânico, fotocópia, de gravação ou outros, somente será permitida com prévia autorização, por escrito, do SENAI.

Esta publicação foi elaborada pela equipe de Educação a Distância do SENAI de Santa Catarina, com a coordenação do SENAI Departamento Nacional, para ser utilizada por todos os Departamentos Regionais do SENAI nos cursos presenciais e a distância.

**SENAI Departamento Nacional**

Unidade de Educação Profissional e Tecnológica - UNIEP

**SENAI Departamento Regional de Santa Catarina**

Gerência de Educação

**SENAI**

Serviço Nacional de  
Aprendizagem Industrial  
Departamento Nacional

**Sede**

Setor Bancário Norte • Quadra 1 • Bloco C • Edifício Roberto  
Simonsen • 70040-903 • Brasília – DF • Tel.: (0xx61) 3317-  
9001 Fax: (0xx61) 3317-9190 • <http://www.senai.br>

# Lista de Ilustrações

---

|  |    |
|--|----|
| Figura 1 - <i>Screenshot</i> do VisualG – Top 10 Livros sem estrutura de dados ..... | 10 |
| Figura 2 - Estrutura de dado unidimensional – Prateleira de Livros .....             | 11 |
| Figura 3 - Estrutura de Dado Bidimensional – Estante de Livros .....                 | 11 |
| Figura 4 - Máquina de Venda de Bebida – Aplicação de Vetor .....                     | 13 |
| Figura 5 - Máquina de Venda de Produtos – Aplicação de Matriz .....                  | 13 |



# Sumário

---

|                              |    |
|------------------------------|----|
| Estruturas de Dados .....    | 9  |
| Apresentação .....           | 9  |
| Definição .....              | 9  |
| Estrutura .....              | 11 |
| Aplicação na indústria ..... | 12 |
| Exemplos .....               | 12 |
| Palavra do Docente.....      | 15 |
| Referências.....             | 16 |





# Estruturas de Dados



## APRESENTAÇÃO

Olá! Bem-vindos aos estudos de Estruturas de Dados!

Agora, você terá a oportunidade de estudar um conceito muito interessante, que, apesar de ser desafiador para a maioria dos iniciantes da programação, é extremamente proveitoso e muito utilizado no universo das soluções computacionais. Trata-se, portanto, do tema Estrutura de Dados.

Neste contexto, será explorado para que é utilizada a Estrutura de Dados, seus principais tipos de representação, técnicas de ordenação, busca, exemplos e aplicações no mundo real.

Bons estudos!

## DEFINIÇÃO

Uma estrutura de dados é uma estratégia para guardar determinadas coleções de dados, sejam eles primitivos ou não. Uma boa forma de entender o que são estrutura de dados é imaginar justamente “coleções”. Na vida real, é comum as pessoas, ao longo de suas vidas, colecionarem determinadas “coisas”. Neste sentido, é possível destacar diversos exemplos de coleções:

- a) Moedas;
- b) Canecas;
- c) Gibis;
- d) Miniaturas de carros.

Mas, o que fazer para guardar uma coleção de um determinado tipo de dado? Será que seria necessário criar uma enorme quantidade de variáveis para cumprir esse papel? Imagine que alguém gostaria de criar um algoritmo capaz de armazenar os seus 10 livros favoritos. Se não tivesse disponível os recursos de estruturas de dados, muito provavelmente seria necessário criar um código com diversas variáveis para armazenar tais valores, algo como:



```
1 Algoritmo "Top 10 Livros"
2
3 Var
4     livroUm : caractere
5     livroDois : caractere
6     livroTres : caractere
7     livroQuatro : caractere
8     livroCinco : caractere
9     livroSeis : caractere
10    livroSete : caractere
11    livroOito : caractere
12    livroNove : caractere
13    livroDez : caractere
14 Inicio
15     escreva("Qual o título do Livro Favorito 01?")
16     leia(livroUm)
17     escreva("Qual o título do Livro Favorito 02?")
18     leia(livroDois)
19     escreva("Qual o título do Livro Favorito 03?")
20     leia(livroTres)
21     escreva("Qual o título do Livro Favorito 04?")
22     leia(livroQuatro)
23     escreva("Qual o título do Livro Favorito 05?")
24     leia(livroCinco)
25     escreva("Qual o título do Livro Favorito 06?")
26     leia(livroSeis)
27     escreva("Qual o título do Livro Favorito 07?")
28     leia(livroSete)
29     escreva("Qual o título do Livro Favorito 08?")
30     leia(livroOito)
31     escreva("Qual o título do Livro Favorito 09?")
32     leia(livroNove)
33     escreva("Qual o título do Livro Favorito 10?")
34     leia(livroDez)
35 Fimalgoritmo
```

Figura 1 - Screenshot do VisualG – Top 10 Livros sem estrutura de dados  
Fonte: do Autor (2020)

Certamente, ao olhar para esse algoritmo, você observou uma repetição de dados. Isso acontece justamente porque não se utilizou nenhuma estrutura de dados para gerenciar essa coleção de livros. Na ausência desse importante recurso, foi necessário criar um “diversas variáveis” com uma declaração similar e um mesmo tipo de dado, sendo modificado apenas o nome do identificador. Esse procedimento, obviamente, não é uma boa prática. O correto seria utilizar as estruturas de dados.

## ESTRUTURA

Duas das estruturas de dados mais básicas são vetores e matrizes. O vetor é uma estrutura de dados com apenas uma dimensão (unidimensional). Já uma matriz é uma estrutura de dados com duas (bidimensional) ou mais dimensões. Observe o exemplo a seguir.

- a) O Vetor seria uma única **prateleira** capaz de guardar determinados objetos, um do lado do outro.
- b) A Matriz bidimensional seria uma grande **estante** com vários espaços, um do lado do outro e um abaixo do outro.

### Vetor de livros



Figura 2 - Estrutura de dado unidimensional – Prateleira de Livros  
Fonte: do Autor (2020)

Veja que, em uma estante de livros (Vetor de Livros), cada livro existente na coleção é posicionado um ao lado do outro, estando limitados a apenas uma dimensão, como se todos os objetos estivessem organizados em uma mesma "linha", nesse caso, da esquerda para a direita sempre.

E no caso de uma matriz? Veja, na imagem, a seguir, uma forma de armazenamento bidimensional, que poderia, também, guardar uma coleção de livros.

### Matriz de livros

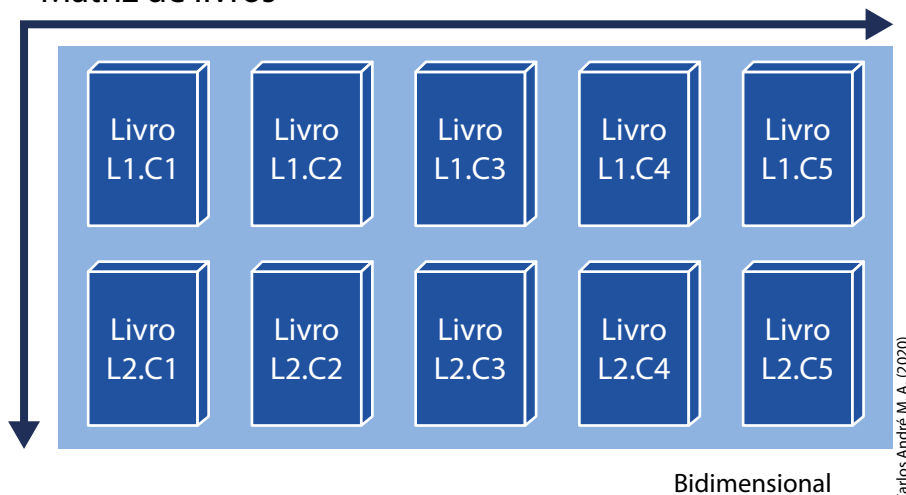


Figura 3 - Estrutura de Dado Bidimensional – Estante de Livros  
Fonte: do Autor (2020)

Observe que, no caso da estante, agora há duas dimensões na estrutura de dados: uma dimensão horizontal (da esquerda para a direita) e uma vertical (cima para baixo). Note que é possível inclusive mapear os livros conforme essas dimensões:

- **Livro L1.C1:** Livro na Linha 1 e Coluna 1;
- **Livro L1.C2:** Livro na Linha 1 e Coluna 2;
- **Livro L1.C3:** Livro na Linha 1 e Coluna 3;
- **Livro L1.C4:** Livro na Linha 1 e Coluna 4;
- **Livro L1.C5:** Livro na Linha 1 e Coluna 5;
- **Livro L2.C1:** Livro na Linha 2 e Coluna 1;
- **Livro L2.C2:** Livro na Linha 2 e Coluna 2;
- **Livro L2.C3:** Livro na Linha 2 e Coluna 3;
- **Livro L2.C4:** Livro na Linha 2 e Coluna 4;
- **Livro L2.C5:** Livro na Linha 2 e Coluna 5.

## APLICAÇÃO NA INDÚSTRIA

No universo prático, em que cenários é possível aplicar as **estruturas** de Dados? A resposta é simples: praticamente em todas as soluções computacionais para o setor produtivo. É muito difícil construir uma boa proposta de programa computacional que não utilize o recurso de estruturas de dados, pois é comum, nos requisitos da solução, que exista a necessidade de trabalhar com coleções de dados.

Ao visitar uma indústria, é fato que ela mesma, em seus aspectos produtivos, trabalha em diversas linhas de produção e seus produtos são concluídos a partir de uma montagem sequencial e linear. E, quando se pensa na capacidade produtiva de vários itens sendo montados ao mesmo tempo, em linhas paralelas de produção, é possível imaginar uma produção matricial.

## EXEMPLOS

Um bom exemplo para ilustrar o tema matrizes é o da Máquina de Venda de Produtos, normalmente encontrada em *Shopping Centers*. Essas máquinas poderiam tanto ser vetores como matrizes de produtos.



Figura 4 - Máquina de Venda de Bebida – Aplicação de Vetor

No caso de uma máquina de venda de bebidas, os botões de seleção das bebidas estão organizadas em linha, sequencialmente em uma única dimensão, ou seja, em um vetor.



WesAbrams ([20--?])

Figura 5 - Máquina de Venda de Produtos – Aplicação de Matriz

Já no caso de uma máquina de venda de produtos, como o da ilustração, os produtos estão organizados em várias prateleiras, em uma distribuição que pode ser entendida como uma matriz bidimensional.



**SAIBA  
MAIS**

Exercite mais a sua compreensão em exemplos que podem ser considerados vetores ou matrizes, pois essa capacidade é muito importante para um bom desenvolvedor de programas computacionais. É comum encontrar a necessidade de se criar uma estrutura de dados e ter que decidir por utilizar vetores ou matrizes.

Acompanhe alguns exemplos e pratique essa classificação:

- a) Uma sala de aula, com diversas carteiras de alunos;
- b) As prateleiras de exposição de produtos de um supermercado;
- c) Uma fila de atendimento de um Banco;
- d) Um acesso para entrada de carros em um estacionamento.

\*\*\*

## PALAVRA DO DOCENTE

---

Compreender os tipos de estruturas de dados é um passo importante para compreensão de como os dados podem ser encontrados, acessados e modificados dentro de um sistema. É um importante passo para o desenvolvedor.

Espero que tenha gostado!

## REFERÊNCIAS

---

ARAÚJO, Everton Coimbra de. **Algoritmos**: fundamentos e prática. Florianópolis: Visual Books, 2005.

SOUZA, Marco Antonio de; GOMES, Marcelo Marques; SOARES, Márcio José; CONCILIO, Ricardo. **Algoritmos e lógica de programação**. São Paulo (SP): Thomson Pioneira, 2005.











*Serviço Nacional de Aprendizagem Industrial*

**PELO FUTURO DO TRABALHO**