1. O que é estrutura de dados
   1. estrutura de dados é uma estrutura organizada de dados na memória de um de computador, de forma que os dados possam ser utilizados de forma correta.
   2. As estruturas se encontram muitas aplicações no desenvolvimento de sistema sendo que algumas são altamente especializadas se utilizada em tarefas específicas.
   3. Usando as estruturas adequadas através de algoritmos, podemos trabalhar com uma grande quantidade de dados, como aplicações em bancos de dados ou serviços de busca.
2. Vetores e Matrizes
   1. Vetores e matrizes são estruturas de dados simples que podem auxiliar quando há muitas variáveis do mesmo tipo em um algoritmo.
3. Registro

Algoritmo

Um algoritmo é o conjunto de instruções estruturadas e ordenadas, seu objetivo é realizar uma tarefa ou uma operação específica.

Os algoritmos são utilizados para manipular dados nas estruturas de várias formas, como por exemplo: inserir, excluir, procurar e ordenar dados.

Em uma estrutura de dados devemos saber como realizar um determinado conjunto de operações básica como por exemplo:

* Inseri Dados
* Excluir Dados
* Localizar um elemento percorrer todos os itens constituintes da estrutura da estrutura para visualização
* Classificar, o que se resume em colocar os itens de dados em uma determinada ordem (numérica, alfabética, etc.)

Principais estruturas de dados

Vetores e Matrizes (tambem chamados de **arrays)**

Registro

Lista

Pilha

Fila

Árvore

Tabela Hash

Grafos

Usaremos o Portugal como IDE para Exemplos

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

Victor ou Array unidimensional é uma variável que armazena várias variáveis do mesmo tipo.

O vetor é uma estrutura de dados indexada, que pode armazenar uma determinada quantidade de valores do mesmo tipo.

1. programa {
2. funcao inicio() {
3. inteiro numeros[] = {39, 45, 54, 55}
5. escreva(numeros[0])
6. }
7. }
8. programa {
9. funcao inicio() {
10. inteiro numeros[] = {39, 45, 54, 55}
12. para (inteiro posicao = 0; posicao <=3; posicao++){
13. escreva(numeros[posicao], " ")
14. }
15. }
16. }
17. programa
18. {
19. funcao inicio()
20. {
21. inteiro vetor[] = { 1, 3, 5, 7, 9} // Cria o vetor com valores pré-definidos
22. inteiro numero
23. logico achou = falso // Variável para armazenar o resultado da procura
25. escreva ("Qual número deseja procurar? ")
26. leia (numero)
28. para (inteiro posicao = 0; posicao < 5; posicao++)
29. {
30. se (vetor[posicao] == numero)
31. {
32. escreva ("Encontrado na posição: ", posicao, "\n")
33. achou = verdadeiro
34. }
35. }
37. se (nao achou)
38. {
39. escreva ("O número não está no vetor\n")
40. }
41. }
42. }

Matriz

Matriz ou array multi-dimensional é um vetor de vetores.

Uma matriz é um vetor que possui duas ou mais dimensões.

Ex: Matriz 4x3

|  |  |  |  |
| --- | --- | --- | --- |
| 2 | -1 | 3 | 1 |
| 0 | -2 | 5 | 4 |
| -3 | 1 | 0 | 0 |

Vetor de 4 posições  
 vetor de 3 posições

1. programa
2. {
3. inclua biblioteca Util --> u
5. funcao inicio()
6. {
7. // Define as dimensões (linhas e colunas) da matriz
8. const inteiro TAMANHO = 5
10. // Cria a matriz
11. inteiro matriz[TAMANHO][TAMANHO]
13. para (inteiro linha = 0; linha < TAMANHO; linha++)
14. {
15. para (inteiro coluna = 0; coluna < TAMANHO; coluna++)
16. {
17. matriz[linha][coluna] = u.sorteia(1, 9) // Atribui um valor aleatório à posição da matriz
19. escreva("[", matriz[linha][coluna], "]") // Exibe o valor contido na posição da matriz
20. }
22. escreva ("\n")
23. }
24. }
25. }

[6][5][4][1][6]

[4][4][9][5][5]

[6][6][9][7][6]

[9][3][4][3][4]

[6][4][5][7][2]

Programa finalizado.