Lógica de Programação – Vetores e Matrizes

O grande detalhe é que independente da linguagem de programação uma vez que se entenda como vetores e matrizes funcionam, a sua aplicação em um código-fonte qualquer que seja a linguagem se torna mais fácil e intuitiva, pois a sintaxe que qualquer linguem pode ser pesquisada rapidamente na internet, se o conceito estiver claro na mente do programador, vetor e matriz deixará de ser esse bicho de 7 cabeças.

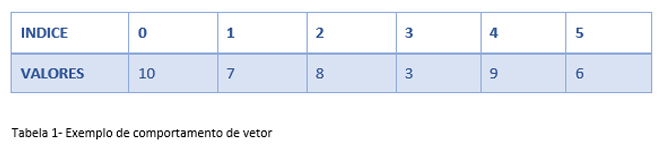
[](http://nerdsti.com.br/wp-content/uploads/2015/11/CAPA.jpg)

Definição de vetor

O vetor pode ser considerado uma variável multidimensional indexada, ou seja, uma variável que tem a capacidade de guardar mais de um valor simultaneamente do mesmo tipo.

Quando falamos de vetor temos que ter a seguinte visão, uma variável comum consegue guardar apenas um valor por vez, enquanto um vetor consegue armazenar diversos valores cada um em uma posição diferente.

Ao observar a tabela 1 podemos entender como um vetor funciona, por exemplo o valor 10 está localizado na posição 0.

[](http://nerdsti.com.br/wp-content/uploads/2015/11/tabela1.jpg)

Vamos usar como exemplo uma situação que seria adequada a utilização do vetor, em uma escola de informática onde o professor passou seis avaliações durante o bimestre entre provas e trabalhos. Para gerar um programa que leia essas seis avaliações e realize a média sem a utilização do vetor em português estruturado ficaria da seguinte forma:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37 | Algoritmo Notas    var    nota1, nota2, nota3, nota4, nota5, nota6, media: real    inicio    escreva(“Digite a nota 1:”)    leia(nota1)    escreva(“Digite a nota 2:”)    leia(nota2)    escreva(“Digite a nota 3:”)    leia(nota3)    escreva(“Digite a nota 4:”)    leia(nota4)    escreva(“Digite a nota 5:”)    leia(nota5)    escreva(“Digite a nota 6:”)    leia(nota6)    media <- (nota1+nota2+nota3+nota4+nota5+nota6)/6;    escreva (“A média é: ”,media)    fim |

Sem a utilização do vetor teríamos que gerar um código desnecessário, agora imagine se fosse 50 notas, a codificação ficaria inviável, antes de realizar o mesmo exemplo citado a cima com vetor, vamos entender como declarar um vetor em português estruturado.

Sintaxe vetor

A sintaxe o vetor não difere muita coisa de uma variável comum, observe:



|  |  |
| --- | --- |
| 1 | vetorNotas: vetor[6] inteiro |

A declaração inicia com o nome do vetor que pode ser qualquer nome, seguido de dois pontos, a palavra vetor e a quantidade de posições que esse vetor terá, em seguida informar o tipo do vetor.

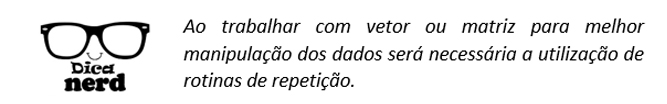
Outros exemplos:



|  |  |
| --- | --- |
| 1  2  3 | vetorNotas: vetor[6] real    nome: vetor[6] texto(50) // comporta em cada posição textos de 50 caracteres. |

Exemplificando

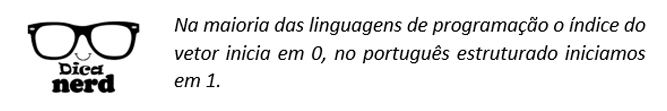
Utilizando o exemplo anterior agora vamos aplica-lo fazendo alterações para implementar o vetor.

[](http://nerdsti.com.br/wp-content/uploads/2015/11/dicaNerd1.jpg)



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | Algoritmo Notas    Var    vetorNotas: vetor[6] real    soma, media: real    contador: inteiro    inicio    soma <- 0    para contador de 1 até 7 passo 1 faça    escreva(“Digite a nota”, contador, “:”)    leia vetorNotas[contador]    soma <- soma + vetorNotas[contador]    fimpara    media <- soma/6;    escreva (“A média é: ”,media)    fim |

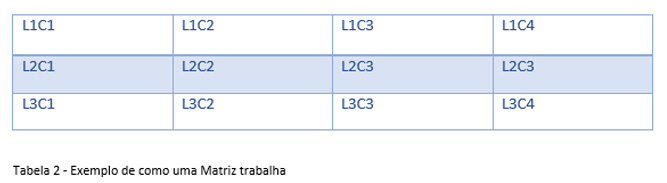
Observe que a codificação ficou bem enxuta e pode ser alterada facilmente caso seja necessário.

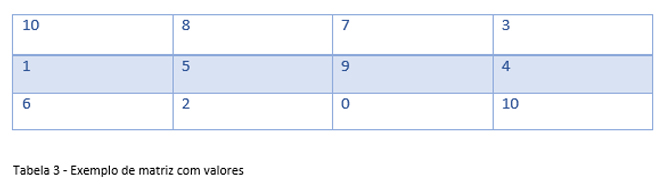
[](http://nerdsti.com.br/wp-content/uploads/2015/11/dicaNerd2.jpg)

Definição de matriz

Assim como o vetor a matriz também é uma variável multidimensional indexada, o seu diferencial é que somente na matriz é possível trabalhar com colunas e linhas formando assim a matriz.

Uma matriz de três linhas e quatro colunas ficaria da seguinte maneira, M[3,4]:

[](http://nerdsti.com.br/wp-content/uploads/2015/11/tabela2.jpg)  
Ao observar a matriz representada acima podemos enxergar a linha e a coluna de cada posição, numericamente essa tabela ficaria da seguinte forma:

[](http://nerdsti.com.br/wp-content/uploads/2015/11/tabela3.jpg)

Ao trabalhar com valores numéricos podemos observar que o número 9 por exemplo está na posição L2C3 (Linha 2 Coluna 3). Agora que entendemos como a matriz funciona vamos observar como passar esse conceito para a lógica de programação. Também será necessária a utilização de estrutura de repetição, como temos duas dimensões (linha e coluna) será necessária a utilização de uma estrutura de repetição dentro da outra.

Vamos complementar o exemplo anterior imaginando que em um ano temos 4 bimestres ( 4 colunas) e em cada bimestre será passado 6 trabalhos ( 6 linhas) com base nessas informações montaremos uma matriz 6×4.

Sintaxe matriz

A sintaxe da matriz é bem parecida a do vetor. Observe:



|  |  |
| --- | --- |
| 1 | matrizNotas: matriz [6],[4] real |

A única diferença é que será necessário informar as dimensões da matriz, lembrando que essa poderá ser bidimensional e até mesmo tridimensional.

Exemplificando



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | Algoritmo Notas    Var    matrizNotas: matriz[6],[4] real    linha, coluna: inteiro    inicio    para linha de 1 até 7 passo 1 faça                    para coluna de 1 até 5 passo 1 faça    escreva(“Digite a”, linha, “ª nota do ”,coluna, “º Bimestre :”)    leia matrizNotas[linha],[coluna]    fimpara    fimpara    fim |

Com esse exemplo simples é possível preencher toda a matriz com seis notas por 4 bimestres. Também é possível implementar a somatória da média por bimestre e anual ( soma dos 4 bimestres), ou até mesmo usar estruturas condicionais para fazer questionamentos se o aluno está aprovado ou reprovado por exemplo.

Considerações finais

Como podemos observar trabalhar com vetores ou matrizes não é nenhum bicho de sete cabeças, é tão simples quanto utilizar uma variável qualquer, o grande detalhe está em utilizar as estruturas de repetição e observar a ordem e posicionamento dos códigos, pois se uma linha for colocada em um lugar errado poderá causar transtornos como armazenamento nas posições erradas ou não conseguir exibir as informações como deseja.

Sendo assim podemos nos atrever a dizer que na maioria das vezes o problema pode ser ocasionado por lógica de programação do que pela implantação do próprio vetor ou da matriz fazendo muitos estudantes ficar temeroso em utilizar tais estruturas tão benéficas.

Espero que tenham gostado desse post e não deixe de curtir para valorizar nosso trabalho e compartilhar com seus amigos que ainda possui uma má impressão sobre o vetor ou matriz.