



SUMÁRIO

1. Objetivo	1
2. Base Teórica	1
2.1. O Tipo Vetor em C	1
2.2. O Tipo Matriz em C	1
3. Sequência da Prática	2
4. Exercício	4

1. OBJETIVO

Introduzir o uso de vetores e matrizes em algoritmos/programas.

2. BASE TEÓRICA

2.1. O Tipo Vetor em C

Um vetor é uma estrutura de dados homogênea (elementos de um mesmo tipo) com apenas uma dimensão (apenas linhas, ou colunas). Um vetor em C é declarado exatamente como uma variável simples, com a diferença de que o tamanho de sua dimensão deve ser definido. Exemplos:

```
int v1[10]; // Vetor de 10 elementos inteiros: v1[0] até v1[9].
float v2[20]; // Vetor de 20 elementos reais: v2[0] até v2[19].
```

Notar que, no C, para um vetor de tamanho n, o primeiro elemento tem índice [0], e o último elemento tem índice [n-1].

Abaixo, um exemplo de uso de vetor. Neste algoritmo/programa, um vetor de tamanho n é gerado onde o valor de cada elemento (v[k]) vale o dobro de seu índice (k), sendo tal vetor impresso.

Fluxograma	Português Estruturado	C
	<pre>início inteiro: k,n,v[10]; imprima("Digite n (<=10): "); leia(n); para k de 1 ate n v[k] ← 2*k; para k de 1 ate n imprima(v[k]); fim.</pre>	<pre>#include "stdio.h" int main(){ int k,n,v[10]; printf("Digite n (n<=10): "); scanf("%d",&n); for (k=0;k<n;k=k+1) v[k] = 2*k; for (k=0;k<n;k=k+1) printf("%d ",v[k]); }</pre>
	Resultado Digite n (n<=10): 5 2 4 6 8 10	Resultado Digite n (n<=10): 5 0 2 4 6 8

2.2. O Tipo Matriz em C

Uma matriz é uma estrutura de dados homogênea (elementos de um mesmo tipo) com duas dimensões (linhas e colunas).

Uma matriz em C é declarada exatamente como uma variável simples, com a diferença de que os tamanhos de suas dimensões devem ser definidos. Exemplos:

```
int m1[5][10]; // Matriz de 5x10 elementos inteiros: m1[0][0] até m1[4][9].
float m2[4][4]; // Matriz de 4x4 elementos reais: m2[0][0] até m2[3][3].
```

Na matriz m[i][j], o primeiro índice, i, se refere à linha i, e o segundo, j, se refere à coluna j. Notar que, no C, para uma matriz m x n, o primeiro elemento tem índices [0][0], e o último elemento tem índices [m-1][n-1].

Abaixo, um exemplo de uso de matriz. Este algoritmo/programa lê uma matriz m x n e faz o somatório dos elementos cuja soma dos índices seja par, imprimindo o resultado.

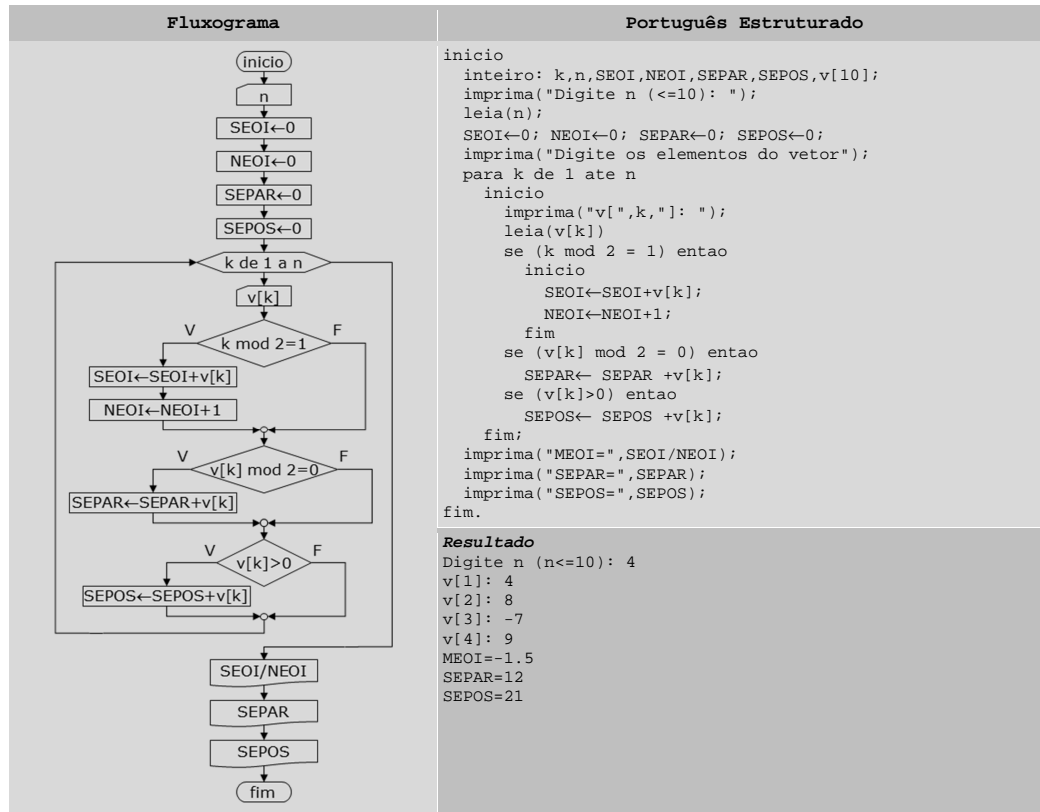
Fluxograma	Português Estruturado
	<pre>início inteiro: i,j,m,n,mat[5,5],s; imprima("Digite m e n (m,n<=5): "); leia(m,n); s ← 0; imprima("Digite os elementos da matriz:"); para i de 1 ate m para j de 1 ate n início imprima("mat[",i,",",j,": "); leia(mat[i,j]); se ((i+j) mod 2 = 0) então s ← s+v[k]; fim; imprima("S=",s); fim.</pre> Resultado Digite m e n (m,n<=2): 2 2 mat[1,1]: 3 mat[1,2]: 7 mat[2,1]: 6 mat[2,2]: 10 S=13

C
<pre>#include "stdio.h" int main(){ int i,j,m,n,mat[5][5],s=0; printf("Digite n (n<=10): "); scanf("%d %d",&m,&n); printf("Digite os elementos da matriz:\n"); for (i=0;i<m;i=i+1) for (j=0;j<n;j=j+1){ printf("mat[%d][%d]: ",i,j); scanf("%d",&mat[i][j]); if ((i+j)%2==0) s=s+mat[i][j]; } printf("S=%d",s); }</pre> Resultado Digite n (n<=10): 2 2 [enter] Digite os elementos da matriz: mat[0][0]: 3 [enter] mat[0][1]: 7 [enter] mat[1][0]: 6 [enter] mat[1][1]: 10 [enter] S=13

3. SEQUÊNCIA DA PRÁTICA

Implementar em C o algoritmo abaixo. Esse algoritmo lê um vetor com N elementos inteiros (N<=10) e calcula o seguinte:

- a média aritmética dos elementos de ordem (índice) ímpar;
- a soma dos elementos que são pares;
- a soma dos elementos positivos.



C
<pre> #include "stdio.h" int main(){ int k,n,SEOI=0,NEOI=0,SEPAR=0,SEPOS=0,v[10]; printf("Digite n (n<=10): "); scanf("%d",&n); printf("Digite os elementos do vetor:\n"); for (k=0;k<n;k=k+1){ printf("v[%k]: ",k); scanf("%d",&v[k]); if (k%2==1){ SEOI=SEOI+v[k]; NEOI=NEOI+1; } if (v[k]%2==0){ SEPAR=SEPAR+v[k]; if (v[k]>0) SEPOS=SEPOS+v[k]; } } printf("MEOI=%.1f SEPAR=%d SEPOS=%d",1.0*SEOI/NEOI,SEPAR,SEPOS); } </pre>
Resultado Digite n (n<=10): 4 [enter] Digite os elementos do vetor: v[0]: 4 [enter] v[1]: 8 [enter] v[2]: -7 [enter] v[3]: 9 [enter] MEOI=8.5 SEPAR=12 SEPOS=21

4. EXERCÍCIO

Dada uma matriz MAT 5 x 4, implemente um programa em C que some os elementos de cada linha e guarde os resultados no vetor SOMALINHA. Em seguida, o algoritmo deve calcular o somatório de todos os elementos da Matriz através da soma dos componentes do vetor SOMALINHA (ou apenas SL), guardando o resultado na variável TOTAL (ou apenas TOT). O somatório de cada linha e o somatório total devem ser impressos.

As expressões para cada um destes somatórios são as seguintes:

$$\text{SomaLinha}_k = \sum_{j=1}^4 \text{MAT}_{kj}, k=1,2,3,4,5$$

$$\text{Total} = \sum_{k=1}^5 \text{SomaLinha}_k$$

O português estruturado desse algoritmo é mostrado ao lado.

Português Estruturado
<pre> inicio inteiro: i,j,TOT,SL[5],M[5,5]; TOT←0; imprima("Digite os elementos da matriz"); para i de 1 ate 5 inicio SL[i]←0; para j de 1 ate 4 inicio imprima("M[",i,"",j,"]: "); leia(M[i,j]) SL[i]←SL[i]+M[i,j]; fim fim imprima("SL[",i,""]=",SL[i]); TOT←TOT+SL[i] fim fim imprima("TOTAL=",TOT); fim. </pre>

Na figura ao lado é mostrado um exemplo da aplicação do algoritmo, onde a matriz MAT é inicializada com determinados valores, o total de cada linha é calculado, e o total de toda a matriz é obtido através da soma dos totais de cada linha.

	SomaLinha				
	1	2	3	4	↓
1	1	6	1	-2	→ 6
2	0	-1	0	2	→ 1
3	1	4	1	4	→ 10
4	-2	2	9	3	→ 12
5	1	-4	1	2	→ 0
TOTAL:					29