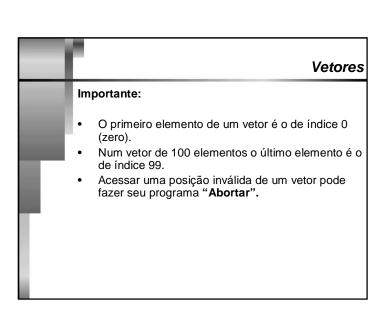
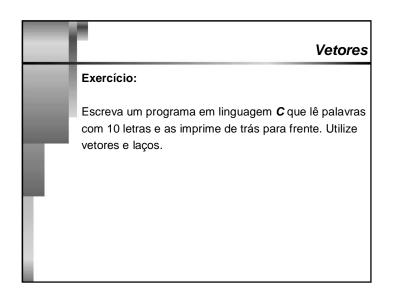


## Vetores Em Linguagem C: tipo + identificador + [ número de variáveis ]; Onde: • tipo: é o tipo das variáveis que devem ser criadas. Ex.: int, char, float, entre outros; • identificador: é o nome que será utilizado para referenciar o conjunto de variáveis; • número de variáveis: é o número de variáveis que será criado (tamanho do vetor).



# vetores Exemplos: int notas[10]; char letras[50]; float pesos[100]; notas[0] = 1; scanf("%c", &letras[6]); notas[5 + 2] = notas[5] \* pesos[2]; for(i = 0; i < N; i++) pesos[i] = 1.0;</pre>



```
Vetores

Solução:

#define TAM 10
int main()
{
    char letras[TAM];
    int i;
    printf("Entre com a frase: ");
    for(i = 0; i < TAM; i++)
        scanf("%c", &letras[i]);
    for(i = TAM - 1; i >= 0; i--)
        printf("%c", letras[i]);
    return 0;
}
```

```
Vetores

Solução:

#define TAM 100
int main()
{
    float notas[TAM];
    int ras[TAM], aux, indice = 0, total;
    printf("Entre com o RA (ou negativo para sair): ");
    scanf("%d", &aux);
    while(aux > 0)
    {
        ras[indice] = aux;
        printf("Entre com a nota: ");
        scanf("%f", &notas[indice]);
        printf("Entre com o RA (ou negativo para sair): ");
        scanf("%d", &aux);
        indice++;
    }
    return 0;
}
```

## Exercício: Escreva um programa em linguagem C que lê RA's e as notas de no máximo 100 alunos. O programa deve ler e armazenar um novo RA e uma nova nota até que o usuário digite um RA negativo.

### Exercício: Utilizando o código do programa anterior, escreva um novo programa que permite ao usuário consultar a nota de um aluno digitando o seu RA, o programa deve rodar até que o usuário digite um RA negativo. Obs.: O trecho de código que lê os RA's e as notas é o mesmo do programa anterior.

###