

Aula Passada

Problema :

Escreva um programa em linguagem **C** que lê os RA's e 4 notas de no máximo 100 alunos. O programa deve ler e armazenar um novo RA e as suas notas até que o usuário digite um RA negativo.

Aula Passada

Solução:

```
#define TAM 100
int main()
{
    float notas1[TAM], notas2[TAM], notas3[TAM], notas4[TAM];
    int ras[TAM], aux, indice = 0, total;
    printf("Entre com o RA (ou negativo para sair): ");
    scanf("%d", &aux);
    while(aux > 0)
    {
        ...
    }
    return 0;
}
```

Matrizes

Problema:

Escreva um programa em linguagem **C** que lê os RA's e ***n*** notas de no máximo 100 alunos, onde ***n*** é definido pelo usuário. O programa deve ler e armazenar um novo RA e as suas notas até que o usuário digite um RA negativo.

Matrizes

Definição:

Uma matriz é uma generalização multidimensional de um vetor, em outras palavras, uma matriz é um vetor de vetores.

Matrizes

Em Linguagem C:

tipo + identificador + [d1][d2] ... [dn] ;

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;
- **d1, d2 ... dn:** são o número de variáveis que serão criados em cada dimensão.

Matrizes

Exemplos :

```
int notas[10][10];
char texto[linhas][colunas];

notas[0][0] = 1;
scanf("%c", &texto[6][0]);
notas[5][2] = notas[5][1] + 1;

for(i = 0; i < N; i++)
    for(j = 0; j < M; j++)
        notas[i][j] = 5.0;
```

Matrizes

Importante:

- Como nos vetores, o primeiro elemento de uma matriz é o de índice 0 (zero).
- No caso de matrizes bidimensionais, a primeira dimensão representa a linha e a segunda a coluna.
- Na memória do computador, matrizes são armazenadas de forma linear como se fossem vetores.
- Durante a compilação o compilador se encarrega de determinar qual o verdadeiro endereço dos campos da matriz.

Matrizes

Exemplo:

A	B	C
D	E	F
G	H	I

A
B
C
D
E
F
G
H
I

Matrizes

Exercício:

Escreva um programa em linguagem **C** que lê os RA's e **n** notas de no máximo 100 alunos, onde **n** é definido pelo usuário. O programa deve ler e armazenar um novo RA e as suas notas até que o usuário digite um RA negativo.

Matrizes

Solução:

```
float notas[TAM][TAM];
int ra[TAM], n, i, j;
printf("Entre com o numero de notas: ");
scanf("%d", &n);
for (i = 0; i < TAM; i++)
{
    printf("Entre com o RA (ou negativo para sair): ");
    scanf("%d", &ra[i]);
    if (ra[i] < 0)
        break;
    for (j = 0; j < n; j++)
    {
        printf("Entre com a nota %d: ", j + 1);
        scanf("%f", &notas[i][j]);
    }
}
```

Matrizes

Exercício:

Escreva um programa em linguagem **C** que solicita ao usuário duas matrizes **A** e **B** de dimensões **n** e **m** (definidas pelo usuário) e imprime a matriz resultante **C** da soma de **A** e **B**.

Matrizes

Solução:

```
int A[TAM][TAM], B[TAM][TAM], C[TAM][TAM], i, j, n, m;
printf("Entre com as dimensoes das matrizes ");
scanf("%d %d", &n, &m);
printf("Entre com os elementos da primeira matriz:\n");
for (i = 0; i < n; i++)
    for (j = 0; j < m; j++)
        scanf("%d", &A[i][j]);
printf("Entre com os elementos da segunda matriz:\n");
for (i = 0; i < n; i++)
    for (j = 0; j < m; j++)
        scanf("%d", &B[i][j]);
for (i = 0; i < n; i++)
    for (j = 0; j < m; j++)
        C[i][j] = A[i][j] + B[i][j];
printf("A matriz resultante eh:\n");
for (i = 0; i < n; i++)
    for (j = 0; j < m; j++)
        printf("%d\t", C[i][j]);
```