

Ex. 0 Escreva um comando **for** que imprime uma sequência de números de zero a 9.

Ex. 1 Escreva um comando **for** duplo (laços aninhados) que imprime a sequência de todos os pares ordenados – o produto cartesiano – de dois conjuntos de dez elementos:

$$(0, 0), (0, 1), (0, 2), \dots (5, 8), (5, 9), (6, 0), (6, 1), \dots (9, 8), (9, 9)$$

Ex. 2 Escreva um comando **for** duplo (laços aninhados) que imprime uma matriz quadrada 10×10 , sendo que o valor do elemento na posição (i, j) é $i + j$:

$$\begin{array}{ccccc} 0 & 1 & \cdots & 8 & 9 \\ 1 & 2 & \cdots & 9 & 10 \\ & & \ddots & & \\ 8 & 9 & \cdots & 16 & 17 \\ 9 & 10 & \cdots & 17 & 18 \end{array}$$

Ex. 3 Escreva um programa, empregando um comando **for**, que imprime a lista de ângulos (em radianos) no intervalo $[0, 2\pi]$, em degraus de $\pi/8$, mais o seno e o cosseno dos respectivos ângulos. Seu programa deve incluir a biblioteca `<cmath>`.

Ex. 4 Para os exercícios abaixo, escreva um programa principal que peça ao usuário para fornecer os elementos dos vetores e exiba os resultados da execução das funções com os valores fornecidos – este é o tipo de programa que *provavelmente* será solicitado em prova. Alternativamente, inicie os vetores no local em que são declarados.

Ex. 5 Reescreva a função que recebe como argumentos um inteiro com o tamanho dos vetores e dois vetores de inteiros, e copia o primeiro vetor para o segundo vetor, na ordem inversa, usando um comando **for**. O protótipo da função é mostrado abaixo.

```
void reverte(int tam, int fte[], int dst[]);
```

Ex. 6 Reescreva uma função que recebe dois vetores de inteiros, soma seus conteúdos elemento a elemento e escreve o resultado num terceiro vetor, usando um comando **for**. O protótipo da função é mostrado abaixo.

```
void soma_vet(int tam, int a[], int b[], int res[]);
```

Ex. 7 Escreva uma função que recebe dois vetores de inteiros, multiplica seus conteúdos elemento a elemento e retorna a soma todos os produtos dos pares de elementos, usando um comando **for**. O protótipo da função é mostrado abaixo.

```
int prod_escalar(int tam, int a[], int b[]);
```