

### **AUTOAVALIAÇÃO 06: estrutura de repetição em C**

Elabore um programa em C que solicita e lê um número inteiro ao usuário e exibe uma mensagem informando se tal número é primo ou não.

Quadro 1: Classificação das dicas.

<b>DICA 1</b>	Apresenta as entradas necessárias, as saídas esperadas e o processamento para transformar as entradas nas saídas de dados.
<b>DICA 2</b>	Aponta a sintaxe das estruturas necessárias para a resolução do problema.
<b>DICA 3</b>	Retrata a lógica de programação indicada para a resolução do problema, em forma de descrição narrativa.

### DICA 1:

**Dados de Entrada:** Um valor inteiro ( $n$ ).

**Dados de Saída:** Mensagem informando se o número ( $n$ ) é primo ou não.

**Processamento:** Ao receber o valor de  $n$ , verificar se o número é primo e, se for, deve-se informar que  $n$  é primo, senão, deve-se informar que  $n$  não é primo. Para verificar se um número é primo, é necessário realizar uma sucessão de divisões de  $n$  por 1 até ele mesmo ( $n$ ) e, ao mesmo tempo, contabilizar a quantidade de números divisíveis encontrada nesse intervalo. Se a quantidade de números divisíveis encontrada for igual a dois,  $n$  é primo; caso contrário,  $n$  não é primo. Portanto, um número só é primo se ele for divisível somente por um e por ele mesmo. Por exemplo: o número 7 só é divisível por 1 e por 7, logo, é primo; já o número 4 é divisível por 1, 2 e 4, logo, não é primo.

### DICA 2:

Esta autoavaliação necessitará de uma estrutura de repetição. Recomenda-se o uso da estrutura de repetição **for**, conforme a sintaxe a seguir:

```
for(<inicialização>; <condição>; <incremento/decremento>){
    <bloco-v>
}
```

Assim como necessitará das estruturas de decisão **if** e **if-else**, conforme as sintaxes a seguir:

```
(i)
if(<condição>){
    <bloco-v>
}
```

```
(ii)
if(<condição>){
    <bloco-v>
}
else{
    <bloco-f>
}
```

**DICA 3:**

Solicitar o valor de  $n$ .

Receber o valor de  $n$ .

Realizar uma sucessão de divisões de  $n$  por 1 até ele mesmo ( $n$ ). Para cada uma dessas divisões, verificar:

Se a divisão for exata:

Incrementar a quantidade de números divisíveis encontrada.

Se a quantidade de números divisíveis encontrada nesse intervalo for igual a dois:

Exibir uma mensagem informando que  $n$  é primo.

Senão:

Exibir uma mensagem informando que  $n$  não é primo.