

O CPF (Cadastro de Pessoas Físicas), emitido pela Receita Federal, é caracterizado por uma função entre o conjunto das pessoas físicas cadastradas e o conjunto dos documentos emitidos.

Ou seja, o fato de um número de CPF ser autenticado pelos seus dígitos verificadores, não o torna um CPF válido, pois é necessário que ele esteja cadastrado no banco de dados da Receita Federal. Assim, um número válido de CPF nem sempre será um documento já emitido. Porém, os dígitos verificadores servem para alertar que o número foi escrito de forma inadequada, sem precisar acessar o banco de dados da Receita Federal.

Regra Prática

O número de um CPF tem 9 algarismos e mais dois dígitos verificadores, que são indicados após uma barra. Logo, um CPF tem 11 algarismos. O número do CPF é escrito na forma ABCDEFGHI/JK ou diretamente como ABCDEFGHIJK, onde os algarismos **não podem ser todos iguais** entre si.

O **J** é chamado **1º dígito verificador** do número do CPF.

O **K** é chamado **2º dígito verificador** do número do CPF.

Primeiro Dígito

Para obter **J** multiplicamos A, B, C, D, E, F, G, H e I pelas constantes correspondentes:

A	B	C	D	E	F	G	H	I	
x10	x9	x8	x7	x6	x5	x4	x3	x2	

O resultado da soma, $10A + 9B + 8C + 7D + 6E + 5F + 4G + 3H + 2I$, é **dividido por 11**.

Analizamos então o **RESTO** dessa divisão:

Se for 0 ou 1, o dígito **J** é [0] (zero). Se for 2, 3, 4, 5, 6, 7, 8, 9 ou 10, o dígito **J** é [11 - RESTO]

Segundo Dígito

Já temos **J**. Para obter **K** multiplicamos A, B, C, D, E, F, G, H, I e J pelas constantes correspondentes:

A	B	C	D	E	F	G	H	I	J	
x11	x10	x9	x8	x7	x6	x5	x4	x3	x2	

O resultado da soma, $11A + 10B + 9C + 8D + 7E + 6F + 5G + 4H + 3I + 2J$, é **dividido por 11**.

Verificamos então o **RESTO** dessa divisão:

Se for 0 ou 1, o dígito **K** é [0] (zero). Se for 2, 3, 4, 5, 6, 7, 8, 9 ou 10, o dígito **K** é [11 - RESTO].

Ex29_CPF_111121

```
static void Main(string[] args)
{
    string CPF, POS; // Variveis de Texto entrada
    int Tamanho, Verificar = 0, Resto = 0, Multiplicador = 11, Soma = 0; // Variveis de Saida
    int[] num = new int[14]; // Vetor inteiro de 14 Indices

    Console.Clear(); // Limpa Tela
    Console.WriteLine("Digite um CPF: "); //Interface 1
    Console.SetCursorPosition(15, 0); // Posição 1

    CPF = (Console.ReadLine()); // Entrada 1
    Tamanho = CPF.Length; // Processo 1

    for (int i = 0; i < Tamanho; i++) // Laço1 Para
    {
        POS = CPF.Substring(i, 1); // Processo 2
        if (POS == "." || POS == "-") // Condicional 1
        {
            num[i] = 0; // Processo 3
        }
        else // Negação Condicional 1
        {
            num[i] = int.Parse(POS) * Multiplicador; // Processo 4
            Multiplicador--; // Processo 5
        }
    }

    for (int j = 0; j < 13; j++) // Laço 2 - Para
    {
        Soma = Soma + num[j]; // Processo 6
    }

    Resto = Soma % 11; //Processo 7
    Verificar = 11 - Resto; // Processo 8

    if (Resto <= 1) Verificar = 0; //Condicional 2 - Processo 9
    if (Verificar == num[13]) // Condicional 3
    {
        Console.WriteLine("CPF: " + CPF + " Válido"); // Saida 1
    }
    else // Negação Condicional 3
    {
        Console.WriteLine("CPF: " + CPF + " Inválido"); // Saida 2
    }
    Console.ReadLine();
}
```