

# FUNÇÕES NO C++



**REUSE • REDUCE • RECYCLE**

**Indicador de Aprendizagem:** Cria procedimentos e funções com passagem de parâmetros, para solução de problemas computacionais para atender um determinado problema.

# FUNÇÕES NO C++

Funções são usadas para criar pequenos pedaços de códigos separados do programa principal.

Em C/C++, quase tudo, é uma função.

**main ( )** é uma função.

**MAIN**

# FUNÇÕES NO C++

Funções são importantes porque elas **retornam valores**, ajudam a **fragmentar o código em partes menores** e **podem ser utilizadas mais de uma vez** no mesmo programa, poupando preciosos minutos de programação e inúmeras linhas de código.



# FUNÇÕES NO C++

## COMO USAR?

Divide-se em três etapas:

1ª Declaração ou protótipo

2ª Chamada ou invocação

3ª Implementação

# FUNÇÕES NO C++

## COMO USAR?

**1ª Declaração ou protótipo:** efetuada antes função `main()` para indicar que uma nova função está sendo criada adicionalmente as já disponibilizadas pela linguagem;

# FUNÇÕES NO C++

## COMO USAR?

**2ª Chamada ou invocação:** efetuada dentro do fluxo de código com intuito de executar a codificação definida para função.

# FUNÇÕES NO C++

## COMO USAR?

**3ª Implementação:** efetuada após a função main ( ) para especificar o código que será executado quando a função for chamada / invocada.

# FUNÇÕES NO C++

## SINTAXE

### // Protótipo da Função

```
tipo_da_funcao nome_da_funcao(parâmetros);
```

### // Chamada da Função

```
nome_da_funcao(parametros);
```

### // Implementação da Função

```
tipo_da_funcao nome_da_funcao (parâmetros) {  
    // Implementação  
}
```



# FUNÇÕES NO C++

Uma declaração de função é também chamada de **protótipo de função**.

A declaração de uma função fornece as informações necessárias para chamar a função: ela fornece o nome da função; mostra o número e tipo de valores que devem ser passados para a função como argumentos; e descreve o valor (se existe) que a função retorna.

# FUNÇÕES NO C++

```
1  #include <iostream>
2  using namespace std;
3  int soma, num1, num2;
4  // Protótipo da Função
5  void somar();
6  main()
7  {
8      setlocale(LC_ALL, "Portuguese");
9
10     cout<<"\n Informe um número: ";
11     cin>>num1;
12     cout<<"\n Informe outro número: ";
13     cin>>num2;
14     somar ();
15     cout<<"\n\n\n";
16 }
17 // Área das Funções
18 // função soma
19 void somar()
20 {
21     soma = num1 + num2;
22     cout<<" \n\n O valor da soma é: "<< soma;
23 }
```

Declara as variáveis globais "**valor\_a**" e "**valor\_b**" que podem ser acessadas por qualquer função (**main()**, **somar()**, etc.)

Declara (protótipo) a função **void somar()** que não possui parâmetros **()** e não retorna valor algum (**void**);

Chama (invoca) a função **somar()** dentro da função **main()**, neste momento fluxo de código é direcionado para implementação da função **somar()** logo abaixo da **main()**;

Implementação da função **somar()**, código executado quando ela é chamada. Ao término da execução o fluxo de código retorna para linha logo abaixo da chamada da função (**return 0**);

# FUNÇÕES NO C++

## VERSÃO 1

```
1  #include <iostream>
2  using namespace std;
3  int soma, num1, num2;
4  // Protótipo da Função
5  void somar();
6  main()
7  {
8      setlocale(LC_ALL, "Portuguese");
9
10     cout<<"\n Informe um número: ";
11     cin>>num1;
12     cout<<"\n Informe outro número: ";
13     cin>>num2;
14     somar ();
15     cout<<"\n\n\n";
16 }
17 // Área das Funções
18 // função soma
19 void somar()
20 {
21     soma = num1 + num2;
22     cout<<" \n\n O valor da soma é: "<< soma;
23 }
```

# FUNÇÕES NO C++

## VERSÃO 2

```
1  #include <iostream>
2  using namespace std;
3  int result, num1, num2;
4  int somar(); // Protótipo da Função
5  main()
6  {
7      setlocale(LC_ALL, "Portuguese");
8      cout<<"\n Informe um número: ";
9      cin>>num1;
10     cout<<"\n Informe outro número: ";
11     cin>>num2;
12     result = somar();
13     cout<<"\n O resultado é: "<<result;
14     cout<<"\n\n\n";
15 }
16 // Área das Funções
17 // função somar
18 int somar()
19 {
20     int soma;
21     soma = num1 + num2;
22     return (soma);
23 }
24
```

# FUNÇÕES NO C++

## VERSÃO 3

```
1  #include <iostream>
2  using namespace std;
3  int num1, num2;
4  int somar(); // Protótipo da Função
5  main()
6  {
7      setlocale(LC_ALL, "Portuguese");
8      cout<<"\n Informe um número: ";
9      cin>>num1;
10     cout<<"\n Informe outro número: ";
11     cin>>num2;
12     cout<<"\n O resultado é: "<<somar();
13     cout<<"\n\n\n";
14 }
15 // Área das Funções
16 // função somar
17 int somar()
18 {
19     int soma;
20     soma = num1 + num2;
21     return (soma);
22 }
```

# FUNÇÕES NO C++

## VERSÃO 4

```
1  #include <iostream>
2  using namespace std;
3  int num1, num2;
4  int somar(); // Protótipo da Função
5  main()
6  {
7      setlocale(LC_ALL, "Portuguese");
8      cout<<"\n Informe um número: ";
9      cin>>num1;
10     cout<<"\n Informe outro número: ";
11     cin>>num2;
12     cout<<"\n O resultado é: "<<somar();
13     cout<<"\n\n\n";
14 }
15 // Área das Funções
16 // função somar
17 int somar()
18 {
19     | return (num1 + num2);
20 }
```

# FUNÇÕES NO C++

## VERSÃO 5

```
1  #include <iostream>
2  using namespace std;
3  int num1, num2;
4  //função inline
5  int somar()
6  {
7      return (num1 + num2);
8  }
9  main()
10 {
11     setlocale(LC_ALL, "Portuguese");
12     cout<<"\n Informe um número: ";
13     cin>>num1;
14     cout<<"\n Informe outro número: ";
15     cin>>num2;
16     cout<<"\n O resultado é: "<<somar();
17     cout<<"\n\n\n";
18 }
19
```

# FUNÇÕES NO C++

## VERSÃO 6

```
1  #include <iostream>
2  using namespace std;
3  int somar(int x, int y); // Protótipo da Função
4  main()
5  {
6      int num1, num2;
7      setlocale(LC_ALL, "Portuguese");
8      cout<<"\n Informe um número: ";
9      cin>>num1;
10     cout<<"\n Informe outro número: ";
11     cin>>num2;
12     cout<<"\n O resultado é: "<<somar(num1,num2);
13     cout<<"\n\n\n";
14 }
15 // Área das Funções
16 // função somar
17 int somar(int x, int y)
18 {
19     return (x + y);
20 }
```



# FUNÇÕES NO C++

## VERSÃO 7

```
1  #include <iostream>
2  using namespace std;
3  int somar(int x, int y); // Protótipo da Função
4  main()
5  {
6      int num1, num2;
7      setlocale(LC_ALL, "Portuguese");
8      cout<<"\n Informe um número: ";
9      cin>>num1;
10     cout<<"\n Informe outro número: ";
11     cin>>num2;
12     cout<<"\n O resultado é: "<<somar(num1,num2);
13     cout<<"\n\n\n";
14 }
15 // Área das Funções
16 // função somar
17 int somar( int num1, int num2)
18 {
19     return (num1 + num2);
20 }
```

# ATIVIDADE PARA HOJE

1. FUP que tenha duas funções: uma para multiplicar e outra para subtrair.

O programa principal deverá ler três números inteiros e eles devem ser passados para as funções.

O resultado deverá ser impresso no programa principal.

2. FUP que leia dois números floats e um caractere referente ao um dos sinais; + - \* / e que devem ser passados para a função.

Conforme o sinal deverá ser realizado o cálculo e retornado o valor para ser impresso no programa principal.