

# Arquitetura de Computadores, Algoritmos, Lógica de Programação e Circuitos Digitais

Danilo Rios  
19/09/2025

# Função

- Está no conteúdo do semestre passado
- O que é?

# Função

- O que é?
  - É um bloco de código
    - Que normalmente se repete durante o código
    - Que possui uma lógica específica
    - Para organizar o código

# Função

- `<tipo de retorno> <nome da função>(<parâmetros da função>) {  
 //comandos  
}`
- Tipos de retorno
  - void
    - Sem retorno
  - Algum tipo de variável

# Função

```
1  #include <stdio.h>
2
3  #define tamanho 5
4
5  void imprimirNumeroInteiro(int numero) {
6      printf("%d\n", numero);
7  }
8
9  int main() {
10
11      int arrayNumero[tamanho];
12      int i;
13      for(i=0;i<tamanho;i++) {
14          printf("Digite um numero: ");
15          scanf("%d", &arrayNumero[i]);
16      }
17
18      printf("Imprimindo os numeros digitados!\n");
19
20      for(i=0;i<tamanho;i++) {
21          imprimirNumeroInteiro(arrayNumero[i]);
22      }
23
24      return 0;
25 }
```

# Função

```
1 #include <stdio.h>
2
3 #define tamanho 5
4
5 void imprimirNumeroInteiro(int numero) {
```

C:\Users\Danilo\Desktop\funç

```
Digite um numero: 1
Digite um numero: 2
Digite um numero: 3
Digite um numero: 4
Digite um numero: 5
Imprimindo os numeros digitados!
1
2
3
4
5
```

```
-----
Process exited after 2.996 seconds with return value 0
Press any key to continue . . .
```

```
21     imprimirNumeroInteiro(arrayNumero[i]);
22 }
23
24 return 0;
25 }
```

Função

```
1  #include <stdio.h>
2
3  void funcao2() {
4      printf("Oi, sou a funcao 2!\n");
5  }
6
7  void funcao1() {
8      printf("Oi, sou a função 1! Vou chamar a funcao2()\n");
9
10     funcao2();
11 }
12
13 int main() {
14
15     printf("Oi, sou a funcao main!\n");
16
17     funcao1();
18
19     funcao2();
20
21     return 0;
22 }
```

Qual vai ser a saída?

```
1 #include <stdio.h>
2
3 void funcao2() {
4     printf("Oi, sou a funcao 2!\n");
5 }
6
7 void funcao1() {
8     printf("Oi, sou a função 1! Vou chamar a funcao2()\n");
9
10    funcao2();
11 }
12
13 int main() {
14
15     printf("Oi, sou a funcao main!\n");
16
17     funcao1();
18
19     funcao2();
20
21     return 0;
22 }
```

C:\Users\Danilo\Desktop\funcao

Oi, sou a funcao main!

Oi, sou a função 1! Vou chamar a funcao2()

Oi, sou a funcao 2!

Oi, sou a funcao 2!

-----  
Process exited after 0.1857 seconds with return value 0  
Press any key to continue . . .



# Função

- Funções sempre retornam para o ponto do código que foram chamadas
- Não!!! Não existe chamar a função `main()` no meio do código

```
1  #include <stdio.h>
2
3  int somarNumeros(int n1, int n2) {
4      return n1 + n2;
5  }
6
7  int main() {
8
9      int numero1, numero2;
10
11     printf("Digite o primeiro numero: ");
12     scanf("%d", &numero1);
13
14     printf("Digite o segundo numero: ");
15     scanf("%d", &numero2);
16
17     printf("%d + %d = %d", numero1, numero2, somarNumeros(numero1, numero2));
18
19     return 0;
20 }
```

```
1  #include <stdio.h>
2
3  int somarNumeros(int n1, int n2) {
4      return n1 + n2;
5  }
6
7  int main()
8  {
9      int
10         Digite o primeiro numero: 1
11         Digite o segundo numero: 2
12         printf("1 + 2 = 3\n");
13         scanf("-----");
14         Process exited after 1.971 seconds with return value 0
15         Press any key to continue . . .
16         scanf("%d", &numero2);
17
18         printf("%d + %d = %d", numero1, numero2, somarNumeros(numero1, numero2));
19
20     return 0;
}
```

# Função

- Como deve ser o código?
  - Pedir para a pessoa digitar 3 números
  - Imprimir o menor número utilizando a função menorNumero para calcular
  - Ter uma função chamada menorNumero que recebe 3 números inteiros como parâmetro e retorna o menor número

Fun

```
1 #include <stdio.h>
2
3 int menorNumero(int n1, int n2, int n3) {
4     if(n1 < n2) {
5         return n1 < n3 ? n1 : n3;
6     } else {
7         return n2 < n3 ? n2 : n3;
8     }
9 }
10
11 int main() {
12
13     int numero1, numero2, numero3;
14
15     printf("Digite o primeiro numero: ");
16     scanf("%d", &numero1);
17
18     printf("Digite o segundo numero: ");
19     scanf("%d", &numero2);
20
21     printf("Digite o terceiro numero: ");
22     scanf("%d", &numero3);
23
24     printf("O menor numero eh: %d", menorNumero(numero1, numero2, numero3));
25
26     return 0;
27 }
```

Fun

```
1 #include <stdio.h>
```

```
2  
3 int menorNumero(int n1, int n2, int n3) {  
4     if(n1 < n2) {  
5         return n1 < n3 ? n1 : n3;  
6     } else {  
7         return n2 < n3 ? n2 : n3;  
8     }  
9 }
```

C:\Users\Danilo\Desktop\fun

```
Digite o primeiro numero: 1  
Digite o segundo numero: 2  
Digite o terceiro numero: 3  
O menor numero eh: 1
```

```
-----  
Process exited after 5.078 seconds with return value 0  
Press any key to continue . . .
```

```
21 printf("Digite o terceiro numero: ");  
22 scanf("%d", &numero3);
```

```
24 printf("O menor numero eh: %d", menorNumero(numero1, numero2, numero3));
```

```
26 return 0;
```

```
27 }
```

# Função

- Como deve ser o código?
  - Pedir para a pessoa digitar um número inteiro positivo
  - Avaliar e imprimir se o número digitado é primo ou não
  - Ficar em loop pedindo para a pessoa digitar enquanto o número for maior ou igual a 2
  - Ter uma função chamada ehPrimo que recebe 1 número inteiro como parâmetro e retorna um char S ou N

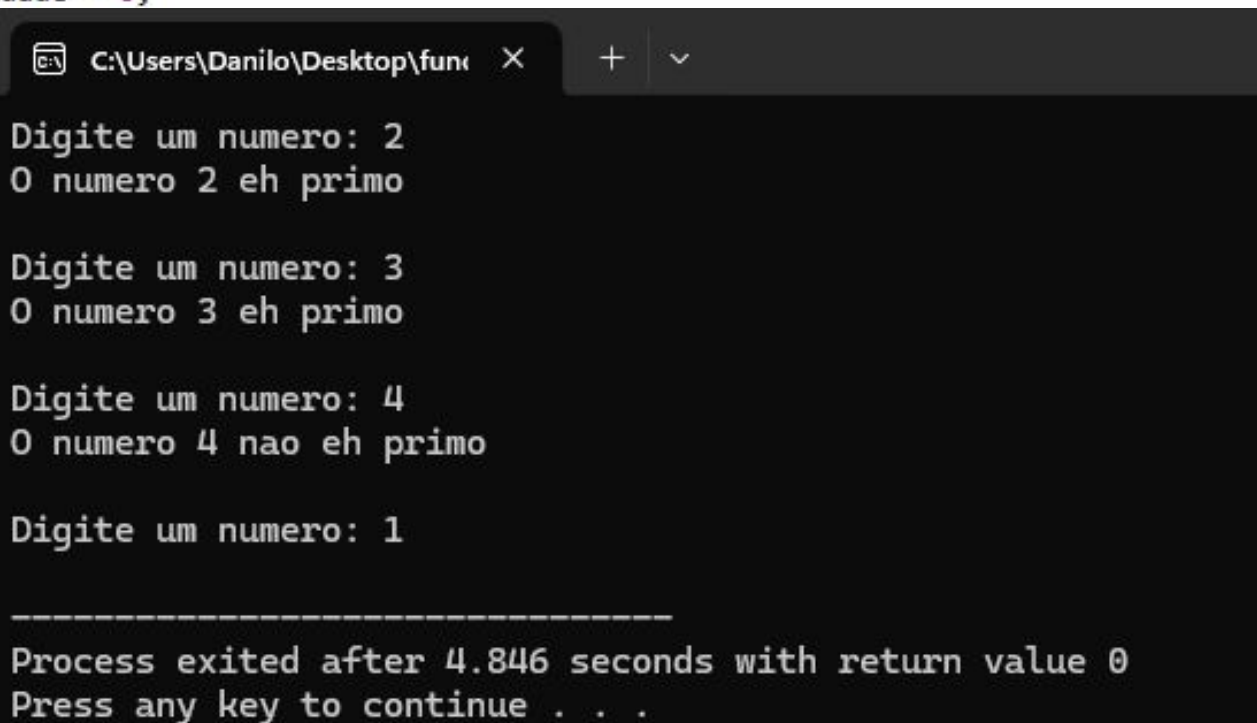
```
1 #include <stdio.h>
2
3 char ehPrimo(int numero) {
4     int i;
5     int quantidade = 0;
6
7     for(i=1;i<=numero;i++) {
8         if(numero % i == 0) {
9             quantidade++;
10        }
11    }
12
13    return quantidade == 2 ? 'S' : 'N';
14 }
15
```

```
16 int main() {
17
18     int numero;
19
20     do {
21         printf("Digite um numero: ");
22         scanf("%d", &numero);
23
24         if(numero >= 2) {
25             if(ehPrimo(numero) == 'S') {
26                 printf("O numero %d eh primo\n\n", numero);
27             } else {
28                 printf("O numero %d nao eh primo\n\n", numero);
29             }
30         }
31
32     } while(numero >= 2);
33
34     return 0;
35 }
```



```
1 #include <stdio.h>
```

```
2  
3 char ehPrimo(int numero) {  
4     int i;  
5     int quantidade = 0;  
6  
7     for(i=1; i<=numero; i++)  
8     {  
9         if(numero % i == 0)  
10            quantidade++;  
11    }  
12  
13    return quantidade < 2;  
14 }  
15
```



```
C:\Users\Danilo\Desktop\fun...  
Digite um numero: 2  
O numero 2 eh primo  
  
Digite um numero: 3  
O numero 3 eh primo  
  
Digite um numero: 4  
O numero 4 nao eh primo  
  
Digite um numero: 1  
-----  
Process exited after 4.846 seconds with return value 0  
Press any key to continue . . .
```

```
31  
32 } while(numero >= 2);  
33  
34 return 0;  
35 }
```

# Função

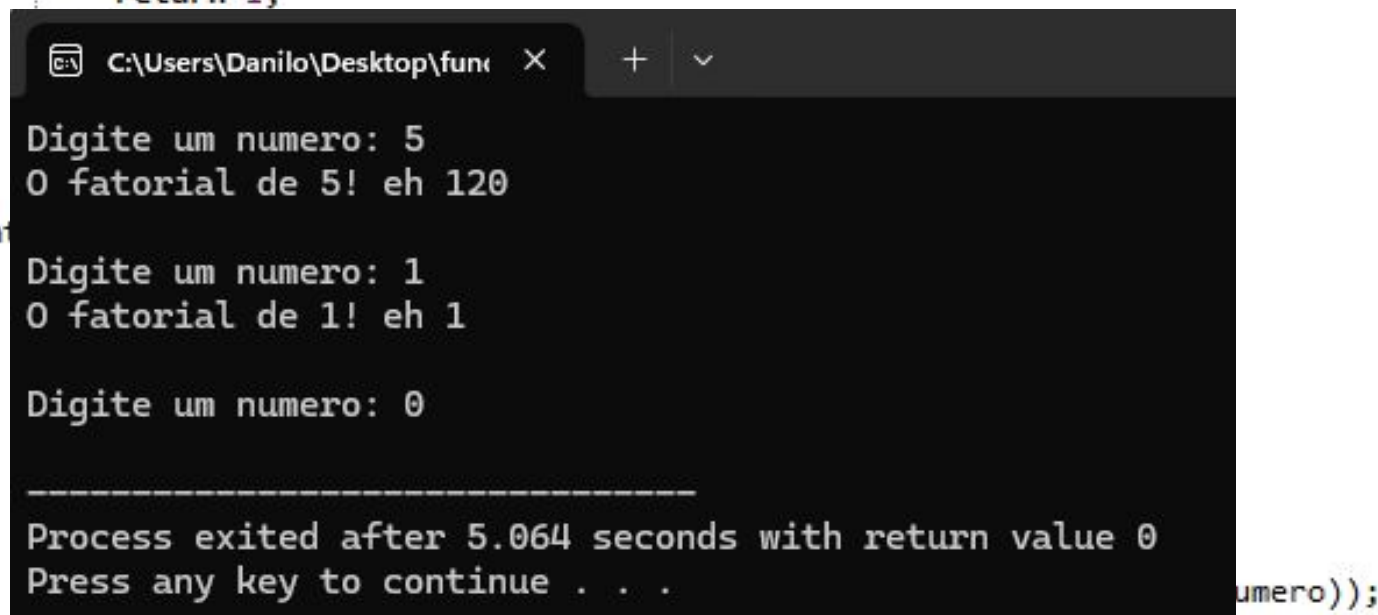
- Como deve ser o código?
  - Pedir para a pessoa digitar um número inteiro positivo
  - Avaliar e imprimir o fatorial do número digitado
  - Ficar em loop pedindo para a pessoa digitar enquanto o número for maior ou igual a 1
  - Ter uma função chamada fatorial que recebe 1 número inteiro como parâmetro e retorna um número inteiro

Fun

```
1  #include <stdio.h>
2
3  int fatorial(int numero) {
4
5      if(numero == 1) {
6          return 1;
7      } else {
8          return numero * fatorial(numero - 1);
9      }
10 }
11
12 int main() {
13
14     int numero;
15
16     do {
17         printf("Digite um numero: ");
18         scanf("%d", &numero);
19
20         if(numero >= 1) {
21             printf("O fatorial de %d! eh %d\n\n", numero, fatorial(numero));
22         }
23
24     } while(numero >= 1);
25
26     return 0;
27 }
```

Fun

```
1 #include <stdio.h>
2
3 int fatorial(int numero) {
4
5     if(numero == 1) {
6         return 1;
7     }
8
9
10 }
11
12 int main()
13 {
14     int numero;
15
16     while(1)
17     {
18         printf("Digite um numero: ");
19         scanf("%d", &numero);
20
21         if(numero < 0)
22             continue;
23
24         printf("O fatorial de %d! eh %d\n", numero, fatorial(numero));
25
26         while(numero >= 1);
27
28         return 0;
29 }
```



```
C:\Users\Danilo\Desktop\func X + v
Digite um numero: 5
O fatorial de 5! eh 120
Digite um numero: 1
O fatorial de 1! eh 1
Digite um numero: 0
-----
Process exited after 5.064 seconds with return value 0
Press any key to continue . . .
```

# Perguntas?

# Exercício 7

- **Obs.: Não será aceito exercício com loop infinito ou chamar a main()**
- Menu com 2 opções
  - Cada opção deve ser uma função
  - 1) Calculadora
    - Menu com 4 opções:
      - Cada opção deve ser uma função
      - +
        - Pedir para digitar 2 números
        - Exibir o resultado  $\langle n1 \rangle + \langle n2 \rangle = \langle resultado \rangle$
        - Voltar para o menu principal
      - -
        - Pedir para digitar 2 números
        - Exibir o resultado  $\langle n1 \rangle - \langle n2 \rangle = \langle resultado \rangle$
        - Voltar para o menu principal
      - \*
        - Pedir para digitar 2 números
        - Exibir o resultado  $\langle n1 \rangle * \langle n2 \rangle = \langle resultado \rangle$
        - Voltar para o menu principal
      - /
        - Pedir para digitar 2 números
        - Exibir o resultado  $\langle n1 \rangle / \langle n2 \rangle = \langle resultado \rangle$
        - Voltar para o menu principal
      - Repetir se não escolher uma opção válida
    - 2) Sair
    - Repetir se não escolher uma opção válida
  - Entrega até: 03/10

# Obrigado!

Até a próxima aula!