

# Exercícios Dia 1

## Esqueleto de um programa em C

```
#include <stdio.h>    //eu sou um comentário de linha

int main ( ) {

    /*
    Eu sou um comentário de bloco
    */

    return 0;
}
```

É dentro da função “main” (principal, em inglês) que você escreverá todo o seu código. A função é chamada de principal porque no futuro, você verá que é possível criar suas próprias funções.

## Tipos de Variáveis

Tipo	O que armazena	Formatação de leitura e escrita	Tamanho na memória	Intervalo de valores
<b>char</b>	Caractere	%c	1 byte	-128 a 127
<b>int</b>	Número inteiro	%d	4 bytes	-2.147.483.648 a 2.147.483.647
<b>float</b>	Número de ponto flutuante (real)	%f	4 bytes	1 , 175494 e-038 a 3,402823 e+038
<b>double</b>	Número de ponto flutuante de precisão dupla	%lf	8 bytes	2,225074 e-308 a 1 ,79693 e+308

float e double são chamados de ponto flutuante porque dentro da memória, um bit é reservado para representar o ponto e ele muda de lugar de acordo com a necessidade. Eles são utilizados para representar números decimais (Ex: 1.09854)

## Funções de leitura e escrita

- scanf(“%d”, &a) : lê um inteiro do teclado e armazena na variável chamada ‘a’
- scanf(“%d %d”, &a, &b) : lê dois inteiros do teclado e armazena nas variáveis ‘a’ e ‘b’
- printf(“%d\n”, a) : imprime a variável ‘a’ na tela. O ‘\n’ faz uma quebra de linha (enter)
- printf(“%d %d\n”, a, b) : imprime as variáveis ‘a’ e ‘b’ na tela

# 1. Aquecendo

1. Reescreva o trecho de código a seguir, corrigindo a indentação:

```
#include<stdio.h>int main(){int saldo = 500;printf("Saldo em Conta: %d", saldo);return 0;}
```

2. Verifique se os códigos a seguir possuem erros. Caso encontre falhas, Corrija:

a) include <stdio.h>

```
int main() [  
    printf("Estou aprendendo! \n");  
    return 0;  
]
```

b) #include <stdio.h> int

```
main() {  
    printf(Está fácil!\n)  
    return 0;  
}
```

c) #include <stdio.h>

```
int main() {  
    int deposito = 0;  
    scanf("%d", deposito);  
    printf("%d\n",  
    deposito);  
    return 0;  
}
```

d) #include <stdio.h>

```
int main() {  
    int divisao = 5 / 3;  
    printf("Resultado = %d", divisao);  
    return 0;  
}
```

## 2. Calculadora

Vamos exercitar a criação de variáveis, receber input do usuário e escrever na tela. Para isso, seu primeiro programa será uma calculadora bem básica. Vamos lá.

Nossa calculadora vai receber sempre dois valores e fazer contas com eles. Para isso, precisamos de, pelo menos, duas variáveis: uma para cada valor. Como queremos trabalhar com números inteiros, qual tipo de variável você deverá criar?

1. Escreva o esqueleto básico de um programa em C
2. Crie duas variáveis do tipo escolhido
3. Peça para o usuário digitar dois números e armazene-os nas variáveis criadas
4. Imprima, uma operação por linha, a soma, a diferença, o produto e o quociente entre os dois números.
5. Compile e execute.
6. Escolha dois números e teste seu programa.
7. Agora vamos arrumar a saída para ficar alinhado: dentro de todos os printf, substitua “%d” por “%2d”. O que aconteceu?
8. Teste sua calculadora com os exemplos abaixo. Brinque um pouco com a sua calculadora inventando outros números.

**Desafio:** Tente fazer com que ela funcione com números decimais.

### Exemplo 1

**Entrada**

4 7

**Saída**

11  
-3  
28  
0

### Exemplo 2

**Entrada**

5 2

**Saída**

7  
3  
10  
2

### 3. Caixa de Padaria

Nossa calculadora era legal, mas... agora queremos criar um caixa para a Padaria Pão Quentinho. Como toda padaria, a Pão Quentinho calcula o seu pão de acordo com o preço por Kg. Note que o uso de casas decimais é de muita importância para uma padaria, pois o peso em gramas influencia no preço total pago pelo pão. Sendo assim, para que o caixa seja eficiente, crie um programa que irá pedir para o caixa da padaria digitar:

- O peso em gramas do pão.
- O valor por quilograma (preço/Kg) do pão.

Como resultado, o programa irá imprimir na tela o custo total da compra de pães com até duas casas decimais de precisão.

**Dica:** Sempre use nome sugestivo para as variáveis. Nesse caso nomear como peso e valor pode ser uma boa pedida.

**Desafio:** Adicione mais 3 produtos à compra, que necessitem de peso e valor/Kg, e faça a soma dos resultados para obter o valor total da compra.

#### Exemplo 1

**Entrada**

10.8 0.9

**Saída**

TOTAL: 9.72

#### Exemplo 2

**Entrada**

12.93 1.245

**Saída**

TOTAL: 16.09

## 4. Conversor de Unidades de Comprimento

O Sistema Internacional de Unidades de Comprimento tem por objetivo sugerir um padrão de medida dessa grandeza ao redor do mundo. Como toda adaptação, é sempre muito trabalhoso converter manualmente as outras formas de comprimento para o padrão desse sistema, que é o metro (m). Para auxiliar nessa Conversão, foi solicitado criar um programa.

1. Nosso conversor fará três conversões em um único programa:
  - a. Polegadas (in) para Metros (m)
  - b. Pés (ft) para Metros (m)
  - c. Milhas (mi) para Metros (m)
2. Para cada conversão, seu programa deve pedir que o usuário digite o comprimento a ser convertido (em número Real). Serão três leituras no total.
3. A impressão deve ser feita com até 4 casas decimais. Serão três impressões no total.

### Estratégias

Seu programa terá que fazer três leituras e três cálculos diferentes. Você tem diferentes estratégias para fazer isso:

- a. Receber as três entradas, calcular e depois imprimir cada uma das saídas
- b. Receber uma entrada por vez, calcular e imprimir o resultado. Fazer isso três vezes.

Qual estratégia você prefere? Fique à vontade para escolher.

### Fórmulas

- Polegadas para Metros:  $m = in * 0.0254$
- Pés para Metros:  $m = ft * 0.3048$
- Milhas para Metros:  $m = mi * 1609,344$

**Dica:** Em alguns casos, vale a pena criar uma variável para armazenar cada resultado. Seu programa pode ficar mais fácil de entender dessa forma.

### Exemplo 1

#### Entrada

23.5 23.5 23.5

#### Saída

0.5969  
7.1628  
37819.58

### Exemplo 2

#### Entrada

10 10 10

#### Saída

0.254  
3.048  
16093.44

## 5. Quadrado

Receba como entrada o valor do lado de um quadrado (como um número real) e imprima na tela:

- A área
- O Perímetro

**Dica:** Você deve gastar seus esforços com lógica, caso não saiba a operação que calcula a área e a que calcula o perímetro, procure na internet.

**Desafio:** Adicionando apenas mais uma variável, torne o programa útil para revelar a área e o perímetro de qualquer retângulo.

### Exemplo 1

Entrada
5.0
Saída
AREA: 25.0 PERIMETRO: 20.0

## 6. Salário

Márcio trabalha como vendedor em uma loja e recebe no final do mês um salário fixo mais uma comissão de 12% sobre o valor total das vendas realizadas. Escreva um programa que leia:

- O salário fixo
- O valor total das vendas realizadas

E imprima na tela o total recebido por Márcio com duas casas decimais

### Exemplo 1

Entrada
1000.00 20000.00
Saída
TOTAL: 3400.00

## 7. Consumo de gasolina

Joãozinho quer calcular e mostrar a quantidade de litros de combustível gastos em uma viagem ao utilizar um automóvel que faz 7 km/L. Faça um programa que receba:

- O tempo gasto na viagem (em horas)
- A velocidade média durante a viagem (em km/h)

Ajude Joãozinho a calcular quantos litros são necessários para fazer a viagem. Imprima seu resultado com precisão de dois dígitos após o ponto decimal.

**Dica:** Observe a correlação entre as grandezas dadas no exercício.

**Desafio:** Supondo que o tanque suporte 45 Litros de gasolina, calcule também quantos tanques serão necessários para a viagem.

### Exemplo 1

Entrada
8.0 60.0
Saída
68.57

## 8. Idade da Dona Astride

Dona Astride, era uma senhora muito vaidosa. Sempre que era questionada sobre sua idade, ela respondia em dias. Pedro, bisneto de Astride, sempre foi um menino muito curioso, mas bem preguiçoso. Ele não queria ter que fazer essa conta de cabeça ou no papel, toda vez que alguém respondesse a idade em dias. Para matar a curiosidade de Pedrinho, faça um programa que calcule a idade de uma pessoa, sempre que forem informados quantos dias ela tem. O programa deve receber:

- O numero de dias que a pessoa tem.

Imprima a quantidade de anos, meses e dias que a pessoa tem (assumindo que o ano tem 365 dias e que os meses **restantes**, sempre terão 30 dias).

**Dica:** Lembre-se de que para fazer esses cálculos você precisará de calcular o resto (mod %) da divisão.

### Exemplo 1

**Entrada**

500

**Saída**

1 ano, 4 meses e 15 dias

### Exemplo 2

**Entrada**

30000

**Saída**

82 anos, 2 meses e 10 dias

## 9. Distância entre dois pontos

Faça um programa que receba dois pontos quaisquer no plano cartesiano, de coordenadas (x, y) e calcule a distância euclidiana entre eles. Dica: a biblioteca math.h contém a função sqrt(), que recebe como parâmetro uma variável double e retorna a raiz quadrada.

- Receba dois valores reais x e y referentes ao ponto a
- Receba dois valores reais x e y referentes ao ponto b

Distância euclidiana:

$$distância = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Imprima a distância com 2 casas decimais.

**Curiosidade:** a distância euclidiana é comumente utilizada em algoritmos de machine learning, ramo da inteligência artificial.

### Exemplo 1

**Entrada**

1 2 3 4

**Saída**

Distancia: 2.83

## 10. Caixa eletrônico

Considerando a existência de notas (cédulas) nos valores R\$ 100, R\$ 50, R\$ 20, R\$ 10, R\$ 5, R\$ 2 e R\$ 1, escreva um programa que receba um valor inteiro e determine o **menor** número de notas para se obter o montante fornecido. O programa deve exibir o número de notas para cada um dos valores de nota existentes.

**Dica:** Você já viu algo que envolveu um calculo parecido anteriormente.

**Desafio:** Calcule também para o saque de centavos (moedas de R\$ 0.50, R\$ 0.25, R\$ 0.10, R\$ 0.05).

### Exemplo 1

#### Entrada

188

#### Saída

1 nota(s) de R\$ 100,00  
1 nota(s) de R\$ 50,00  
1 nota(s) de R\$ 20,00  
1 nota(s) de R\$ 10,00  
1 nota(s) de R\$ 5,00  
1 nota(s) de R\$ 2,00  
1 nota(s) de R\$ 1,00

### Exemplo 2

#### Entrada

8473947

#### Saída

84739 nota(s) de R\$ 100,00  
0 nota(s) de R\$ 50,00  
2 nota(s) de R\$ 20,00  
0 nota(s) de R\$ 10,00  
1 nota(s) de R\$ 5,00  
1 nota(s) de R\$ 2,00  
0 nota(s) de R\$ 1,00