

# Calculadora em C

Trabalho feito por: Jonathan Bryan Kanssaw Gomez

## Primeira parte - Início do Código

O código se dá início nas funções que ela irá utilizar para fazer o cálculo.

Para a Soma, eu defini a variável “adição” como “float” e defini o “Return” com a soma dos valores que serão inseridos posteriormente.

Como na Soma eu fiz o mesmo com os operadores restantes.

```
8
9  #include <stdio.h>
10
11  //Função para Adição
12  float adicao(float a, float b){
13      return a + b;
14  }
15
16  //Função para Subtração
17  float subtracao(float a, float b){
18      return a - b;
19  }
20
21  //Função para Divisão
22  float divisao(float a, float b){
23      return a / b;
24  }
25
26  //Função para Multiplicação
27  float multiplicacao(float a, float b){
28      return a * b;
29  }
30
```

## Segunda Parte - Menu

Esta é a parte do código mais importante: O menu do Usuário.

Primeiro definimos as variáveis do código, "Float" para os dois números reais que serão calculados posteriormente.

Logo depois definimos a variável do operador, definimos ela inicialmente como "5" para que o código "entre" no While.

No While, definimos a verificação que ele irá utilizar para executar a repetição. Enquanto variável "op" for diferente de zero, ele continuará executando a calculadora. E dentro do While escrevemos o Menu que irá aparecer ao usuário

No final se o usuário Digitar 0, o programa irá entrar no "If" e executará o "Return 0", que fará o programa encerrar

```
31 int main(){
32 float num1, num2;
33 int op = 5;
34 while (op != 0){
35     printf("Digite o operador do Cálculo:\n");
36     printf("SAIR ----- 0\nAdição ----- 1\nSubtração ----- 2\n");
37     printf("Multiplicação - 3\nDivisão ----- 4\n");
38     scanf("%d", &op);
39 if(op == 0){
40     return 0;
41 }
```

```
Digite o operador do Cálculo:
SAIR ----- 0
Adição ----- 1
Subtração ----- 2
Multiplicação - 3
Divisão ----- 4
```

## Terceira Parte – O Cálculo.

É nesta parte do cálculo que toda a magia acontece!

Como o usuário já havia digitado o operador desejado no código anterior, nós só precisamos dos números desejados para a operação. Logo nós pedimos ao usuário que digite os dois números desejados e definimos os números como “num1” e “num2”.

E finalmente ao cálculo, para fazermos isso existe várias formas, porém acho “If” o mais simples.

Como o operador já foi selecionado acima, é só usarmos uma verificação para que o programa “entre” no “If” e comece as operações.

Quando a verificação for a correta, ele irá executar o que estiver escrito no programa.

Por exemplo:

Na adição, ele informa os dois números selecionados e o resultado, para que o usuário tenha certeza de que está correto a operação.

E então é escrito “adicao” que é a função que programamos anteriormente, e ele irá colar o que estiver escrito nela.

```
41 printf("Digite dois números:\n");
42 scanf("%f %f", &num1, &num2);
43
44 if (op == 1){
45     printf("%.2f + %.2f = %.2f\n\n\n", num1, num2, adicao(num1, num2));
46 }else if(op == 2){
47     printf("%.2f - %.2f = %.2f\n\n\n", num1, num2, subtracao(num1, num2));
48 }else if (op == 3){
49     printf("%.2f * %.2f = %.2f\n\n\n", num1, num2, multiplicacao(num1, num2));
50 }else if (op == 4){
51     if (num2 == 0){
52         printf("Error: Divisão por 0.\n");
53     }else {
54         printf("%.2f / %.2f = %.2f\n\n\n", num1, num2, divisao(num1, num2));
55     }
56 }
57 } else {
58     printf("Operador inválido.\n");
59 }
60 }
61 return 0;
62 }
```

Fiz a mesma coisa com os outros operadores, exceto a Divisão. Como você já deve saber, zero dividido por zero não existe! Ele dá erro, então para resolver este problema eu adicionei outro "If", para verificar se o segundo número digitado for zero, ele dirá ao usuário: "Error: Divisão por 0.". Mas se for outro número ele irá executar a operação normalmente.

```
50 ~    }else if (op == 4){  
51 ~        if (num2 == 0){  
52 ~            printf("Error: Divisão por 0.\n");  
53 ~        }else {  
54 ~            printf("%.2f / %.2f = %.2f\n\n\n", num1, num2, divisao(num1, num2));
```

Porém se o usuário digitar outro número que não esteja no menu de seleção de operadores, (Ex: 7)

```
Digite o operador do Cálculo:  
SAIR ----- 0  
Adição ----- 1  
Subtração ----- 2  
Multiplicação - 3  
Divisão ----- 4
```

```
57 ~    } else {  
58 ~        printf("Operador inválido.\n");  
59 ~    }  
60 ~ }  
61 ~ return 0;  
62 ~ }
```

O programa dirá: Operador inválido e pedirá ao usuário digitar novamente.

**Operador inválido.**



1º

```

8
9 #include <stdio.h>
10
11 //Função para Adição
12 float adicao(float a, float b){
13     return a + b;
14 }
15
16 //Função para Subtração
17 float subtracao(float a, float b){
18     return a - b;
19 }
20
21 //Função para Divisão
22 float divisao(float a, float b){
23     return a / b;
24 }
25
26 //Função para Multiplicação
27 float multiplicacao(float a, float b){
28     return a * b;
29 }
30

```

3º

```

41 printf("Digite dois números:\n");
42 scanf("%f %f", &num1, &num2);
43
44 if (op == 1){
45     printf("%.2f + %.2f = %.2f\n\n", num1, num2, adicao(num1, num2));
46 }else if (op == 2){
47     printf("%.2f - %.2f = %.2f\n\n", num1, num2, subtracao(num1, num2));
48 }else if (op == 3){
49     printf("%.2f * %.2f = %.2f\n\n", num1, num2, multiplicacao(num1, num2));
50 }else if (op == 4){
51     if (num2 == 0){
52         printf("Error: Divisão por 0.\n");
53     }else {
54         printf("%.2f / %.2f = %.2f\n\n", num1, num2, divisao(num1, num2));
55     }
56 } else {
57     printf("Operador inválido.\n");
58 }
59
60 }
61 return 0;
62 }

```

2º

```

31 int main(){
32     float num1, num2;
33     int op = 5;
34     while (op != 0){
35         printf("Digite o operador do Cálculo:\n");
36         printf("SAIR ----- 0\nAdição ----- 1\nSubtração ----- 2\n");
37         printf("Multiplicação - 3\nDivisão ----- 4\n");
38         scanf("%d", &op);
39     if (op == 0){
40         return 0;
41     }

```

Este é o código completo:

Fim