

[Clique aqui para ler pelo PDF e melhorar a experiência](#)

Funções Básicas de Entrada e Saída em C



Nesta aula, vamos explorar as funções básicas de entrada e saída na linguagem C, focando principalmente em `scanf()` e `printf()`. Estas funções são fundamentais para interagir com o usuário, permitindo que você receba dados e exiba resultados.

Funções de Saída: `printf()`

A função `printf()` é utilizada para exibir texto na tela. Você pode formatar a saída de várias maneiras, incluindo a inserção de variáveis no texto.

Exemplo de uso do `printf()`:

```
#include <stdio.h>

int main() {
    int idade = 30;
    printf("Sua idade eh: %d anos.\n", idade);
    return 0;
}
```

Caso não tenha ficado claro a declaração da variável `idade`, você pode retornar para a segunda aula postada no meu blog.

Neste exemplo, `%d` é um especificador de formato que indica que uma variável do tipo inteiro será inserida no texto. Mas se fosse uma variável do tipo float seria `%f`, para caracteres `%c`.

No código acima, será exibido na tela "Sua idade é 30 anos" e uma quebra de linha.

Uma característica muito interessante do `printf()` é que você pode e em alguns casos deve limitar a quantidade de casas decimais exibidas como pode ser visto abaixo:

```
#include <stdio.h>

int main() {
    float nota1 = 9;
    float nota2 = 8;
```

```
float media = (nota1 + nota2) / 2;
printf("Sua media final eh: %.2f.\n", media);
return 0;
}
```

Perceba que a média, dada pela soma das duas notas e divisão por dois, é printada na aplicação. Porém, prestando bem atenção o formatador da variável no printf() não é somente um %f mas um %.2f, a inclusão de .2 limita a exibição do valor para duas casas decimais. Se fosse %.3f, três casas decimais e assim por diante. Então no caso, pode se afirmar que a saída final do programa será "Sua media final eh: 8.50" somado de uma quebra de linha.

Funções de Entrada: scanf()

A função `scanf()` é utilizada para receber dados do usuário. Ela lê a entrada padrão (geralmente o teclado) e armazena o valor nas variáveis especificadas.

Exemplo de uso do scanf():

```
#include <stdio.h>

int main() {
    int idade;
    printf("Digite sua idade: ");
    scanf("%d", &idade);
    printf("Você tem %d anos.\n", idade);
    return 0;
}
```

Note que usamos o operador & antes da variável idade. Isso é necessário porque scanf() precisa do endereço da variável para armazenar o valor lido.

Não se preocupe quanto a utilização do &, será abordado mais para frente quando estudarmos ponteiros!

No código acima é declarado uma variável idade que será atribuído a ela o valor que o usuário digitar. Se o usuário digitar 20, o programa "printará" no fim "Você tem 20 anos." junto de uma quebra de linha.

Prática com pequenos programas

Muito bem, parabéns por chegar até aqui. Agora chegou a hora de por a mão na massa.

Aqui estão três sugestões de programas simples que você pode implementar para praticar:

1. **Calculadora Simples:** Um programa que recebe dois números flutuantes e realiza a soma, subtração, multiplicação e divisão.

Exemplo de execução:

```
Digite o primeiro numero: 10 --> entrada
Digite o segundo numero: 5 --> entrada
Soma: 15 --> saída
Subtracao: 5 --> saída
Multiplicacao: 50 --> saída
Divisao: 2 --> saída
```

O programa deve se comportar como visto acima, atente-se as quebras de linhas e ao tipo dos dados que estão sendo lidos, nesse caso floats.

2. **Programa de Média:** Um programa que solicita ao usuário três notas (valores não inteiros) e calcula a média.

Exemplo de execução:

```
Digite a primeira nota: 8.5 --> entrada
Digite a segunda nota: 9 --> entrada
Digite a terceira nota: 9.5 --> entrada
A media eh 9.00 --> saída
```

3. **Cálculo de IMC:** Um programa que recebe a altura e o peso do usuário e calcula o Índice de Massa Corporal (IMC).

Exemplo de execução:

```
Digite seu peso em kg: 85.5 --> entrada
Digite sua altura em metros: 1.91 --> entrada
Seu IMC eh: 23.43 --> saída --> saída
```

Sinta-se à vontade para mandar os exercícios ou dúvidas no meu e-mail (ericklaus16@gmail.com).

Conclusão

Parabéns! Agora você conhece as funções básicas de entrada e saída na linguagem C e já pode criar programas interativos simples. No próximo tutorial, vamos explorar os conceitos de estruturas condicionais.

[\[Voltar ao Topo\]](#)