



- Siga-nos no Twitter
- FEED RSS - O que é isso?
- Adicione aos Favoritos

- BLOG
- FÓRUM
- SITEMAP
- QUEM SOMOS
- CONTATO

alizada

Buscar

Enquete

O Google Chrome OS irá desbancar o Microsoft Windows 7?

- ☐ sim
- ☐ não

parciaisvotar

Programação

- Algoritmo
- C/C++
- Java

Web

- HTML/XHTML
- JavaScript
- PHP

Sistema Operacional

- Comandos de DOS
- Windows
- Linux
- Mac/BSD

Office

- Word / Writer
- Excel / Calc

Áreas do Site

- Download
- Fórum
- Blog

Parcerias

Recomendamos

Conheçam

TI Expert » Programação » C/C++ » Tudo Sobre Máscaras de Dados

Máscara de Dados

Máscaras de dados são usadas para definir entradas e saídas de dados em C. Basicamente, ela demonstra ao programa que tipo de conversão deve ser adotado para converter os bytes armazenados em uma variável. Por exemplo, se entrarmos com uma variável de ponto flutuante (float) que usa 4 bytes de memória, na saída, também devemos utilizar uma máscara de ponto flutuante. Se utilizarmos uma máscara de número inteiro, o algoritmo vai ler apenas os dois primeiros bytes, pois uma variável inteira tem apenas a metade dos bytes de ponto flutuante. E além disso, variáveis int, float e double usam padrões diferentes de armazenamento. Por exemplo, se digitarmos a entrada 10, ela será convertida em bytes em int, que será convertida em bytes diferentes em float, que por sua vez, será totalmente diferente de double. Apenas char é igual int e vice-versa.

```
01. /* EXEMPLO DO QUE PODE ACONTECER DE ERRADO EM UM PROGRAMA
02. Aqui declararemos uma variavel inteira valendo 65.
03. veja o que acontece quando mandamos imprimir o inteiro como flutuante
04. */
05.
06. #include <stdio.h>
07.
08. int main (void)
09. {
10.     int nr=65;
11.     printf ("Este e um int\t-> %d\n", nr);
12.     printf ("Este e um char\t-> %c\n", nr);
13.     printf ("Este e um float\t-> %f\n", nr);
14.     return 0;
15. }
```

Como podemos ver, apenas int e char deram um resultado satisfatório, float resultou em algo inesperado - nada.

As máscaras

Agora que vimos para que servem as máscaras, veremos qual é a máscara de cada tipo de entrada e saída.

máscara	tipo de dado	descrição
%d	int	mostra um número inteiro
%c	char	mostra um caracter
%f	float ou double	mostra um número decimal
%i	int	mostra um número inteiro
%ld	long int	mostra um número inteiro longo
%e	float ou double	mostra um número exponencial (número científico)
%E	float ou double	mostra um número exponencial (número científico)
%o	int	mostra um número inteiro em formato octal
%x	int	mostra um número inteiro em formato hexadecimal
%X	int	mostra um número inteiro em formato hexadecimal
%s	char	mostra uma cadeia de caracteres (string)

```
01. // Exemplo de mascaras e formato de seus dados
02.
03. #include <stdio.h>
04.
05. int main (void)
06. {
07.     printf ("Mascara d\t-> %d\n", 1);
08.     printf ("Mascara c\t-> %c\n", 'A');
09.     printf ("Mascara f\t-> %f\n", 1.65);
10.     printf ("Mascara i\t-> %i\n", 1);
11.     printf ("Mascara ld\t-> %ld\n", 1);
12.     printf ("Mascara e\t-> %e\n", 1.65);
13.     printf ("Mascara o\t-> %o\n", 256);
14.     printf ("Mascara x\t-> %x\n", 256);
15.     printf ("Mascara s\t-> %s\n", "Cadeia de caracteres (string)");
16.     return 0;
17. }
```

Alguns cuidados

O problema de usar máscaras é a ambiguidade que pode trazer ao compilador. Por exemplo: se por acaso quisermos utilizar o símbolo %, como faremos? Nesse caso podemos usar outros tipo de máscara.

máscara	mostra
%%	%
\"	" (aspas)
\'	' (aspas simples)

Também podemos usar modificadores nas máscaras

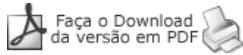
mascara	modificação	descrição
%d	%+d	mostra o sinal de positivo se houver
%f	%."número"f	o número que vier entre o ponto e o f será o limite de sua precisão
%o	%#o	completa o número octal com zero
%x	%#x	mostra indicador de Hexadecimal (0x).

```
01. // Exemplo de modificadores de mascaras
02.
03. #include <stdio.h>
04.
05. int main (void)
06. {
07.     printf ("Original \"%d\"\nModificado: %+d %d\n\n", 1, 1, -1);
08.     printf ("Original \"%f\"\nModificado: %.1f %.3f\n\n", 1.2345, 1.2345, 1.2345);
09.     printf ("Original \"%o\"\nModificado: %#o\n\n", 256, 256);
10. }
```

```
10. | printf ("Original \"%x\"\\nModificado: %#x\\n", 256, 256);  
11. | }
```



Autor: [Denys William Xavier](#)  
Este artigo está sob Licença [Creative Commons](#).

[< Funções de Saída em C](#)**C/C++**[Funções de Saída em C++ >](#)

Faça o Download  
da versão em PDF



Gostou?  
Indique nosso site!

Este artigo foi  
lido **65251** vezes



Compartilhe

### Páginas Relacionadas

[C/C++ - Estrutura de Dados - STRUCT](#)  
[C/C++ - Declaração de Variáveis](#)  
[C/C++ - Typedef - Definição de Nomes de Tipos](#)  
[C/C++ - Processos e Atribuição de Valor a Variáveis](#)  
[C/C++ - Typecast \(Conversão de Tipos de Dados\)](#)  
[Java - O que são Atributos e Variáveis](#)  
[Java - Tipos de Dados](#)  
[Java - Enum](#)  
[Java - Typecast e Instanceof - Indução de Tipos](#)  
[Excel/Calc - Formato de Dados em uma Planilha](#)

» **Programação** > Algoritmo | C/C++ | Java

» **Web** > HTML/XHTML | JavaScript | PHP

» **Sistema Operacional** > Comandos de DOS | Windows | Linux | Mac/BSD

» **Office** > Word/Wirter | Excel/Calc

» **Áreas do Site** > Downloads | Fórum | Blog

» Quem Somos

» Contato

» Termos de Uso

WWW.TIEXPERT.NET © 2007 - 2010