

Lista de especificadores (format specifiers)

A tabela a seguir traz os formatos mais comuns utilizados

Formatador	Descrição
%c	Para o tipo de caractere.
%d ou %i	Para o tipo inteiro
%f ou %lf	Para o tipo float (com 6 casas decimais)
%l ou %L	Para o tipo longo acrescido o <tipo>
%ld ou %li	Para o tipo inteiro longo
%lf	Para o tipo double
%g ou %G	Para o tipo float/double com precisão
%e ou %E	Para notação científica (<valorReal>e+00)
%u	Se dado um valor negativo, mostra o primeiro complemento
%o	Representação octal
%x ou %X	Representação hexadecimal
%p	Ponteiro - mostra aonde está o valor na memória
%u	Representação para unsigned value (valor não sinalizado)

O formatador para a representação binária não existe em C

```
#include<stdio.h>
/*Codigo para verificar a quantidade de bytes alocados
para cada tipo utilizado em um programa/computador
*/
int main()
{
    char charType;
    double doubleType;
    float floatType;
    int intType;
    // Tamanho em bytes dos tipos mais usados em variaveis
    printf("Tamanho em bytes do tipo char...: %lu \n", sizeof(charType));
    printf("Tamanho em bytes do tipo int...: %lu \n", sizeof(intType));
    printf("Tamanho em bytes do tipo float.: %lu \n", sizeof(floatType));
    printf("Tamanho em bytes do tipo double: %lu \n", sizeof(doubleType));
    return 0;
}
```

```
#include<stdio.h>
// Exemplo sobre inicializacao de variaveis e parametros do printf
int main()
{
    int a=0,b=1;
    printf("Digite dois numeros inteiros separados por espaco: ");
    // a linha seguinte sera' mostrada apos a entrada dos valores
    // recebidos com scanf
    printf("O scanf retorna um valor, que deve ser 2 em caso de sucesso: %d \n",
scanf("%d %d",&a,&b));
    printf("Os valores ZERO e UM sao apenas uma inicializacao para as variaveis.\n
nos valores recebidos do usuario foram: %d %d",a,b);
    return 0;
}
```

Tabela ASCII

<https://www.ascii-code.com>