

## Algumas funções importantes do C

O que estiver entre < > (maior e menor) indica o tipo do parâmetro ou variável que vai no local onde esses símbolos aparecem.

<char\*> indica que neste local deve vir uma variável do tipo string (char[])  
<int> indica que neste local deve vir uma variável do tipo int  
<char> indica que neste local deve vir uma variável que armazena só uma letra  
<void\*> indica que neste local você deve colocar o endereço (&) de uma variável de qualquer tipo.  
<void> indica que neste local não vai variável.

Por exemplo:

<int> getch(<void>)  
Indica que o getch() retorna um número inteiro.

scanf(<char\*>, <void\*>)  
Indica que o scanf() tem como parâmetros uma string (representado pelo char\*), que indica o formato da variável a ser lida, e um ponteiro (endereço) para uma variável qualquer de memória (indicado pelo void\*).

Funções de entrada de dados (para utilizá-las, inclua a biblioteca conio.h):

gets(<char\*>) gets(string)  
Lê uma string do teclado e coloca na variável string (char\*) indicada como parâmetro.  
Exemplo: char nome[20];  
gets(nome);

<int> getch(<void>) <variável> = getch()  
Função que lê uma tecla pressionada pelo usuário (não é necessário que o usuário pressione [Enter] depois de ter pressionado a tecla). O código da tecla pressionada é retornado pela função como um número inteiro <int> e pode ser colocado em uma variável. A tecla que o usuário digitou não é mostrada na tela.  
Exemplo: char tecla;  
tecla = getch();

<int> getche(<void>) <variável> = getch()  
Faz a mesma coisa que o getch(), porém, a tecla pressionada pelo usuário é mostrada na tela.  
Exemplo: char tecla;  
tecla = getche();

Funções de formatação de dados (para utilizá-las, inclua a biblioteca stdio.h):

sprintf(<char\*>, <char\*>, <void\*>) sprintf(<destino>, <formato>, <origem>)  
Função idêntica ao printf, só que, ao invés de imprimir na tela, imprime em uma string (vetor de caracteres) indicada pela variável destino.  
Exemplo: char destino[20];  
float valor=10.1;  
sprintf(destino, "5.2f", valor); // destino = 10.10

sscanf(<char\*>, <char\*>, <void\*>) sscanf(<origem>, <formato>, &<destino>)  
Função idêntica ao scanf, só que, ao invés de ler o valor do teclado, ele é lido de uma string indicada pela variável origem.  
Exemplo: char origem[] = "9.2";  
float numDestino;  
sscanf(origem, "05.2f", &numDestino); // numDestino = 09.20

Funções de formatação de caracteres (para utilizá-las, inclua a biblioteca ctype.h):

<int> toupper(<int>) <letraMaiuscula> = toupper(<letraMinuscula>)  
Converte uma letra o caractere minúsculo em maiúsculo.  
Exemplo: printf("%c", toupper(getch()));

<int> tolower(<int>) <letraMinuscula> = tolower(<letraMaiuscula>)  
Converte uma letra ou caractere maiúsculo para minúsculo.  
Exemplo: char tecla, maiuscula;  
tecla = getch();  
maiuscula = tolower(tecla);

