

Funções ou sub-rotinas

Função são comandos agrupados que poderão ser utilizados várias vezes em um programa. A um conjunto de funções é dado o nome de biblioteca, por exemplo a **libc** da linguagem **C** que contém os principais funções da linguagem.

As funções fornece modularidade ao programa, possibilitando a divisão deste em programas menores, este tipo de função são chamadas de funções de usuário, pois é o próprio programador que irá implementá-las.

Exemplo de funções:

Linguagem			Significado
Pascal	C	Python	
write ou writeln	printf	print	Saída de informação, geralmente vídeo
readln	scanf	input	Entrada de informação, geralmente teclado
cos	cos	cos	Calcula o cosseno de um ângulo em radianos

Parâmetros

Parâmetros são variáveis que passamos para as funções que necessitem. As funções podem ter entre zero (0) a vários parâmetros, e em algumas linguagens pode possuir condições de especificar parâmetros opcionais.

Os parâmetros podem ser de 3 tipos distintos, de entrada, de saída ou de entrada e saída.

Os parâmetros de entrada são as variáveis que a função vai necessitar para poder operar.

Os parâmetros de saída podem armazenar dados necessários para retornar para o programa principal.

Os parâmetros de entrada e saída, une as duas formas anteriores.

Valores de retorno

Geralmente as funções retornam o que foi calculado, como por exemplo as funções trigonométricas, que passamos como parâmetro o ângulo em radianos e elas nos devolvem o resultado e há funções que não retornam nada.

O programador desenvolverá as funções de acordo com suas necessidades.

Exemplos:

Linguagens			Significado
Pascal	C	Python	
write ('Ola Mundo');	printf ("Ola Mundo");	print('Ola Mundo');	Mostra na tela a frase "Ola Mundo"
read(nome);	scanf("%s",&nome);	nome = input();	Lê do teclado uma cadeia de caracteres e o resultado será colocado na variável nome
c = cos(angulo);	c = cos(angulo);	c = cos(angulo);	Calcula o cosseno da variável ângulo e o armazena na variável c

Algumas funções básicas para utilização em algoritmos e fluxograma:

Legenda: **valores obrigatórios**, **valores opcionais**

Função	Descrição
Mostrar(arg1 , arg2 , ..., argN)	Mostra na saída padrão os valores das variáveis arg1 até argN
Ler(arg1 , arg2 , ..., argN)	Lê da entrada padrão os valores e estes são colocados nos argumentos de arg1 até argN
Comprimento(cadeiacaracteres)	Retorna o tamanho da cadeiacaracteres .
Comprimento(vetor)	Retorna a quantidade (tamanho) de um vetor
sin(ângulo)	Retorna o seno do ângulo em radianos
cos(ângulo)	Retorna o cosseno do ângulo em radianos
tan(ângulo)	Retorna a tangente do ângulo em radianos
mínimo(arg1 , arg2 , ..., argN)	Retorna o menor valor entre todos os argumentos
máximo(arg1 , arg2 , ..., argN)	Retorna o maior valor entre todos os argumentos
Contém(arg , cadeiacaracter)	Retorna -1 se arg não existir em cadeiacaracter , ou a posição inicial de arg dentro da cadeiacaracter
Randomico(arg1 , arg2)	Gera um número aleatório <= arg1 ou um valor aleatório entre arg1 e arg2 , caso o segundo parâmetro seja dado
Repetir(caracter, N)	Repete o caractere na tela N vezes
Évazio(arg)	Retorna Verdadeiro se o arg estiver vazio ou false se contiver algo
Énúmero(arg)	Retorna verdadeira se o arg for uma variável numérica, caso contrário falso
Életra(arg)	Retorna verdadeira se o arg for uma variável caractere, caso contrário falso
caracter2value(caractere)	Retorna o valor numérico do caractere

value2character(num)	Retorna o caractere correspondente ao valor numérico
----------------------	--

1. Função MINIMO
2. Parametros: **arg1**, **arg2** tipo inteiro
3. Declarar **r** tipo inteiro
4. **r** ← **arg1**
5. Se **arg1** > **arg2** Então
 1. **r** ← **arg2**
6. fim se
7. retorna **r**
8. fim MINIMO

- 1) Programa PRINCIPAL
- 2) Declare **a**, **b** tipo inteira
- 3) mostrar “Digite dois valores inteiros”
- 4) Leia **a**,**b**
- 5) Mostrar “O menor valor digitado foi: “, minimo(**a**,**b**)
- 6) fim PRINCIPAL

8)

- 1) Função CONTADINHEIRO
- 2) Parametros: moeda e nota tipo vetor
- 3)
- 4)

N-1) retorna total

N) Fim CONTADINEHEIRO