

Funções em C/C++

Function Declaration

D CLASSROOM

The diagram illustrates the components of a function signature: `float findArea(float length, float width);`. Labels with arrows point to each part: 'return type' points to `float`; 'function name' points to `findArea`; 'parameter type' points to the first `float` and the second `float`; 'parameter name' points to `length` and `width`; and 'terminating semicolon' points to the semicolon at the end.

dvclassroom.com

Definição

Função é uma subrotina, a qual executa um bloco de códigos e pode retornar um resultado. Uma função pode referenciar a si mesma, o que é chamado de **recursividade**. Em algumas literaturas, quando uma função não retorna um resultado, ela denomina-se procedimento. Em Programação Orientada a Objetos as funções são chamadas de métodos.

Conceito

Ao desenvolver um semáforo, provavelmente você implementará as mesmas instruções para cada lâmpada: acender, aguardar um determinado tempo, apagar. Expressar essas etapas em um código de linguagem de programação 3 vezes custará mais código, porém, escrevendo uma função que realize esses procedimentos, o programa se torna menor e mais legível.

Conceito

Funções são utilizadas a fim de **diminuir a redundância de instruções** em um código-fonte, diminuindo o seu tamanho e melhorando a sua interpretação, tanto pelo seu implementador quanto pelas pessoas que poderão acessá-lo ou alterá-lo.

Formato de uma função

```
tipo_de_retorno nome_da_função (lista_de_parâmetros) {  
    // bloco de comandos  
}
```

- Tipo de retorno: determina qual a natureza do valor que a função retornará.
- Nome da função: um nome, definido com os mesmos padrões para declaração de identificadores e que remete à ação que a função executará.
- Lista de parâmetros: os valores de entrada que serão copiados ou referenciados para dentro da função.

Características

- Nome: segue o mesmo padrão de nomes dos identificadores, ou seja, inicia com letra ou *underline*, seguido de caracteres alfanuméricos ou *underline*. Determine o nome com o menor tamanho possível, mas que remeta à ação que a função realiza.
- Tamanho: normalmente uma função ocupa, com algumas exceções, no máximo 20 linhas do código-fonte.

Características

- Níveis de abstração: uma função tende a ocultar detalhes do programa principal, permitindo com que o desenvolvedor não necessite saber exatamente como, mas apenas o que ela faz.
- Assinatura: a forma de identificar uma função, o que pode ser por meio do seu tipo de retorno, nome ou parâmetros.

Passagem de Parâmetro por Valor

```
#include <stdio.h>
```

```
#include <math.h>
```

```
float circle_area (float radius) {  
    return 3.1415 * pow(radius, 2.0);  
}
```

```
int main(void) {  
    float radius = 2.3;  
    printf("Área do círculo: %.2fm².\n", circle_area(radius));  
    return 0;  
}
```


Passagem de Parâmetros por Referência

```
#include <stdio.h>
```

```
void sum (int *a, int *b) {  
    *a += *b;  
}
```

```
int main(void) {  
    int a = 3, b = 2;  
    printf("A = %i. B = %i.\n", a, b);  
    sum(&a, &b);  
    printf("A = %i. B = %i.\n", a, b);  
    return 0;  
}
```

Referências

- SCHILDT, H. C: **Completo e Total**. 3 ed. São Paulo: Makron Books, 1996. 720 p.
- DEITEL, H. M.; DEITEL, P. J. Como Programar em C. 6 ed. São Paulo: Pearson, 2011. 818 p.
- DAMAS, L. Linguagem C. 10 ed. Rio de Janeiro: LTC 2007. 410 p.

Dúvidas



Obrigado...

**Boa semana
para
todos!!!**