

Curso de Tecnologia em Automação Industrial

2º Período

Lógica de Programação

Escrevendo os primeiros programas

Conforme mencionado anteriormente, para criar um programa de computador, precisamos utilizar uma linguagem de programação.

A nível de complexidade, existem dois níveis que definem as linguagens de programação: Alto nível e baixo nível.

Basicamente, quanto mais próximo a linguagem está do alto nível, menos complexa ela é, o ocorrendo o contrário quando uma linguagem se aproxima do baixo nível.

Escrevendo os primeiros programas

Iremos iniciar na programação utilizando a linguagem C.

A linguagem C foi desenvolvida em 1972, a partir de uma linguagem chamada linguagem B.

Tornou-se conhecida por ter sido a linguagem utilizada para desenvolvimento do sistema operacional Unix.

A linguagem C ANSI foi aprovada em 1989.

Escrevendo os primeiros programas

A Linguagem C é uma linguagem de programação estruturada.

Os códigos fonte escritos nessa linguagem podem ser escritos utilizando “estrutura de sequencia”, “estrutura de seleção” e “estrutura de repetição”.

A utilização de sub-rotinas é o outro motivo pelo qual a Linguagem C pertence a esse paradigma, pois o código fonte é estruturado em sub-rotinas.

Escrevendo os primeiros programas

A seguir, sintaxe e estrutura básica do código fonte de um programa escrito em C.

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

```
main(){  
    comando1;  
    comando2;  
}
```

Escrevendo os primeiros programas

Na estrutura apresentada, temos algumas palavras que são comandos da linguagem. Esses comandos serão descritos a seguir:

`#include<stdio.h>`: As linhas iniciadas por `#` são executadas antes do programa ser compilado. O comando `include` diz ao compilador para incluir o arquivo de cabeçalho `stdio.h`. Esse arquivo de cabeçalho contém instruções e informações utilizadas para compilar algumas operações realizadas através do código fonte.

Lógica de Programação

Escrevendo os primeiros programas

`main()`: Comando que deve existir em todos os programas escritos em C. Os parênteses indicam que `main` é um bloco do programa chamado de função. Todo programa em C possui uma ou mais funções. Uma delas é a `main`. A função `main` é a primeira função que será executada quando o programa for executado.

`{}`: Determina um bloco de comandos na linguagem C.

Escrevendo os primeiros programas

Assim é escrito o código fonte de um programa em C. Dentro da estrutura formada pelos comandos mostrados anteriormente, se combinarão variáveis, expressões, testes, mensagens e demais funções do programa.

A esses comandos vistos, dá-se o nome de palavras reservadas. Essas palavras não podem ser utilizadas dentro do programa, fora da sua função, como nome de variáveis, por exemplo.

Existem outros comandos, ou palavras reservadas, que serão vistos no decorrer das aulas.

Escrevendo os primeiros programas

Declaração de variáveis e constantes em um programa em C:

Nas linguagens de programação em geral, ao criar uma variável, deve-se implementar ou declarar, que tipo de dado essa variável irá armazenar.

Como visto anteriormente, podemos trabalhar com diferentes tipos de dados: Numéricos (inteiros e reais), caracteres e lógicos. Logo as variáveis criadas no programa deverão ser compatíveis com o tipo de dado que irá armazenar.

Escrevendo os primeiros programas

Para esse caso, também temos palavras reservadas na linguagem C, que irão determinar que tipo de dado uma variável ou constante irá armazenar, conforme tabela abaixo:

Tipo de dado	Palavra reservada em C
Inteiro	int
Real	float
Caractere	char

Lógica de Programação

Escrevendo os primeiros programas

Sintaxe de declaração de variáveis em C:

tipo nome;

Também podemos criar mais de uma variável:

Tipo nome1,nome2,nome3;

Escrevendo os primeiros programas

Exemplo de declaração de variáveis em C:

```
int valor1;
```

Também podemos criar mais de uma variável:

```
float valor1,valor2,valor3;
```

Lógica de Programação

Escrevendo os primeiros programas

Sintaxe de declaração de constantes em C:

```
int const nome=valor;
```

Sintaxe para declaração de múltiplas constantes:

```
int const nome1=valor,nome2=valor;
```

Lógica de Programação

Escrevendo os primeiros programas

Exemplo de declaração de constantes em c:

```
int const x=10;
```

Também podemos criar mais de uma constante:

```
int const x=10,y=20;
```

Lógica de Programação

Escrevendo os primeiros programas

Além do conceito de variáveis, antes de escrever o primeiro programa, devemos conhecer os conceitos de entradas e saídas.

Conforme mencionado anteriormente, um programa de computador, basicamente, recebe algum dado (entrada), executa sobre ele algum tipo de processamento e devolve um resultado desse processamento (saída).

Essas entradas e saídas devem ser implementadas pelo programador no código fonte do programa.

A seguir, os comandos de entrada e saída utilizados na linguagem C.

Escrevendo os primeiros programas

Comando de saída “**printf**” - Não há sentido, por exemplo, uma calculadora realizar muitos cálculos, se não houver um display para exibir os resultados. Em algum momento, nossos programas deverão imprimir mensagens de instruções na tela, ou exibir resultados de algum processamento.

Essa é a função do comando printf: Imprimir mensagens na tela.

Além de mensagens, o comando também pode imprimir o conteúdo de variáveis na tela, conforme sintaxe e exemplos a seguir:

Lógica de Programação

Escrevendo os primeiros programas

Sintaxe:

```
printf("mensagem");  
printf("%especificador",variável);
```

Exemplos:

```
printf("Informe o valor da primeira nota do aluno ");
```

O comando acima, ao ser executado, irá imprimir a mensagem entre " " na tela.

Escrevendo os primeiros programas

Exemplos:

```
printf("%especificador",media);
```

O comando acima irá imprimir na tela o conteúdo da variável “media” (previamente criada na sessão de declaração de variáveis do código fonte). Não ficará claro para o usuário do programa, se apenas o valor for impresso na tela. O ideal é que seja impressa uma mensagem, dizendo o que é esse valor. Para esse caso, o comando printf pode ser utilizado da seguinte maneira:

```
printf("A média entre os valores informados é %especificador",media);
```

O comando acima irá imprimir na tela, a mensagem entre “ “ e o conteúdo da variável media (previamente criada).

Escrevendo os primeiros programas

Comando de entrada “**scanf**” – Nem todos os dados que um programa manipula são gerados por ele.

Em algum momento o usuário deverá fornecer dados para que o mesmo realize algum tipo de processamento e devolva um resultado.

A sintaxe do comando scanf tem o seguinte formato:

```
scanf(“%especificador”,&variável);
```

O comando scanf irá pausar a execução do programa até que seja informado algo pelo usuário. Quando o dado for informado, ele será guardado na variável especificada após o caractere &.

Escrevendo os primeiros programas

Exemplo:

```
scanf("%especificador",&nota1);
```

O comando acima, ao ser executado, irá fazer uma pausa na execução do programa, até que o usuário informe um valor e pressione a tecla Enter. Ao fazer isso o programa irá guardar o valor informado na variável “nota1” (previamente criada), e prosseguir com a execução dos outros comandos do código fonte. Também pode ser utilizado um único scanf para várias variáveis:

```
scanf("%especificador%especificador",&nota1,&nota2);
```

Lógica de programação

Linguagem C – Especificadores de conversão

Especificador	Significado
%d	Número inteiro
%f	Número real
%c	Um caractere
%s	Uma string
%%	Imprimir o caractere %

Lógica de programação

Funções numéricas, algébricas e trigonométricas:

cos(valor): retorna o cosseno de um ângulo, representado pela variável valor;

pow(base,expoente): retorna o valor da exponenciação de base elevado ao expoente;

sqrt(valor): retorna a raiz quadrada do valor representado pela variável valor;

cbrt(valor): retorna a raiz cúbica do valor representado pela variável valor;

sin(valor): retorna o seno do ângulo representado pela variável valor;

tan(valor): retorna a tangente do ângulo representado pela variável valor;

% (numerador%denominador): Resto da divisão de numerador por denominador.

Escrevendo os primeiros programas

Erros:

Basicamente, dois erros podem acontecer nos programas criados. Erros de sintaxe e erros de execução.

Erros de sintaxe: São os erros que ocorrem por problemas de sintaxe nas regras e comandos da linguagem utilizadas. Nesse caso, ao ser testado o programa não roda e o interpretador informa o erro ocorrido

Erros de execução: Nesse caso os programas rodam, mas não dão uma saída correta ao que foi solicitado. O resultado apresentado deve ser validado. Normalmente isso é feito testando a saída informada através do processo manual automatizado pelo programa.

Escrevendo os primeiros programas

Primeiro programa:

Utilizaremos os conceitos vistos até aqui, utilizando a estrutura de um programa em C, vista anteriormente, com os comandos e conceitos mostrados para criar o primeiro programa.

Para esses primeiros programas, iremos utilizar estruturas sequenciais, ou seja, todos os comandos digitados no código fonte serão executados de forma sequencial, sem desvios, decisões ou repetições.

Enunciado do problema a seguir:

Escrevendo os primeiros programas

1 – Faça um programa que solicite ao usuário informar dois valores numéricos. Após isso, fazer a soma entre os valores e apresentar o resultado dessa soma na tela.

Antes de implementar uma solução, conforme mencionado anteriormente, recomenda-se que o processo a ser automatizado através da criação de um programa seja conhecido.

A seguir solução para o problema proposto. Trata-se do código fonte do programa, utilizando a estrutura vista anteriormente, junto com os conceitos mostrados.

Escrevendo os primeiros programas

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

```
#include <cstdlib>
```

```
main()
```

```
{
```

```
    int n1,n2,s;
```

```
    printf("Digite um número ");
```

```
    scanf("%d",&n1);
```

```
    printf("Digite outro número ");
```

```
    scanf("%d",&n2);
```

```
    s=n1+n2;
```

```
    printf("A soma entre os números é %d \n",s);
```

```
    system("pause");
```

```
}
```

Escrevendo os primeiros programas

2 – Faça um programa que solicite ao usuário informar um número. Apresentar na tela o valor anterior, posterior e o dobro do número informado.

Escrevendo os primeiros programas

```
#include <stdio.h>
#include <cstdlib>
main()
{
    int n;
    printf("Digite um numero ");
    scanf("%d",&n);
    int a,p,d;
    a=n-1;
    p=n+1;
    d=n*2;
    printf("O numero anterior e %d \n",a);
    printf("O numero posterior e %d \n",p);
    printf("O dobro e %d \n",d);
    system("pause");
}
```

Escrevendo os primeiros programas

3 – Faça um programa que solicite ao usuário informar o seu nome e apresente uma mensagem de boas vindas ao mesmo.

Escrevendo os primeiros programas

```
#include <stdio.h>
#include <cstdlib>
main()
{
    char n[30];
    printf("Digite seu nome ");
    scanf("%s",&n);
    printf("Ola %s, seja bem vindo \n",n);
    system("pause");
}
```

Exercícios

- 1 – Faça um programa que leia o nome de um aluno e as notas das três provas que ele obteve no semestre. No final informar o nome do aluno e a sua média (aritmética);
- 2 – Faça um programa que calcule o consumo médio de um automóvel, sendo fornecida a distância percorrida e o total de combustível gasto;
- 3 - Faça um programa que leia o nome de um vendedor, o seu salário fixo e o total de vendas efetuadas por ele no mês (em dinheiro). Sabendo que este vendedor ganha 15% de comissão sobre suas vendas efetuadas, informar o seu nome, o salário fixo e salário no final do mês.
- 4 - Ler dois valores para as variáveis A e B, e efetuar as trocas dos valores de forma que a variável A passe a possuir o valor da variável B e a variável B passe a possuir o valor da variável A. Apresentar os valores trocados.

Exercícios

5 – Faça um programa que calcule o valor de delta de uma equação do segundo grau, lendo os parâmetros a, b e c. A fórmula para cálculo do delta é $\Delta = B^2 - 4AC$. Após calcular o valor do delta, calcular os valores das raízes da equação.

6 – Faça um programa que receba uma temperatura informada em graus Celsius e apresente a mesma convertida em graus Fahrenheit. Fórmula para conversão: $F = (9 \times C + 160) / 5$;

7 - Faça um programa que converta um valor informado em reais (R\$), calcule e apresente a sua conversão para dólar e euro. O valor da cotação dessas moedas deverá ser informado;

8 – Faça um programa que receba um valor que foi depositado em uma poupança, calcule e exiba o valor com rendimento após um, dois, três, quatro, cinco e seis meses. Considere fixo o juro da poupança de 0,7 ao mês;

9 – Uma loja está vendendo em até 5x, com juros de 2% ao mês. Faça um programa que receba o valor de uma compra, a quantidade de parcelas, calcule e exiba o valor das prestações;

10 – Faça um programa que solicite a idade de um indivíduo e informe a quantidade de dias e meses vividos pelo mesmo. Considere um ano igual a 365 dias e um ano igual a 12 meses.

Exercícios

- 11 – Faça um programa que receba uma temperatura informada em graus Fahrenheit e apresente a mesma convertida em graus Celsius. Fórmula para conversão: $C = ((F - 32) * 5) / 9$;
- 12 – Faça um programa que calcule e apresente o valor do volume de uma caixa retangular utilizando a fórmula $VOLUME = COMPRIMENTO * LARGURA * ALTURA$;
- 13 – Faça um programa que receba um número e apresente o resultado do número elevado ao quadrado;
- 14 – Faça um programa que receba o valor da base e do expoente de uma potência, calcule e informe o resultado do cálculo;
- 15 – Faça um programa que receba três valores e apresente como resultado a soma dos quadrados dos três valores;
- 16 – Em uma eleição sindical concorreram ao cargo de presidente três candidatos. Faça um programa que receba a quantidade de votos de cada candidato, a quantidade de votos nulos e brancos, calcule e informe o percentual de votos dos três candidatos em relação ao total de eleitores e o percentual de votos nulos e brancos em relação ao total de eleitores;
- 17 – Faça um programa que calcule a raiz de uma base qualquer com índice qualquer.