

Introdução a Programação

Conceitos e Comandos Básicos

Hebert Coelho

Instituto de Informática
Universidade Federal de Goiás

Roteiro

- Variáveis
- Constantes
- Operadores
- Entrada de Dados

Variáveis

- Uma *variável* é uma posição nomeada de memória, que é usada para guardar um valor que pode ser modificado pelo programa.
- Todas as variáveis devem ser declaradas antes de serem usadas.
- Variáveis serão declaradas em três lugares básicos: dentro de funções, na definição de parâmetros das funções e fora de todas as funções.

Forma geral de uma declaração:

```
tipo lista_de_variáveis;
```

Variáveis

Exemplos:

```
1 int i, j, l;  
2 double var, cont, valor;
```

Inicialização de variáveis

- Pode ser dado um valor à variável no mesmo momento em que elas são declaradas, colocando um sinal de igual e uma constante após o nome da variável.

Forma geral:

tipo nome_da_variável = constante;

Exemplos:

```
1 int x = 0;  
2 float altura = 1.60;
```

Relembrando

printf

A função printf() tem a seguinte forma geral:

```
printf(string_de_controle, lista_de_argumentos);
```

String de controle

Código	Significado
<code>%d</code>	inteiro
<code>%f</code>	float
<code>%c</code>	caractere

Segundo Programa

Escrevendo inteiros

```
1 #include <stdio.h>
2 int main(){
3     int idade = 15;
4     printf("%d",idade);
5     return(0);
6 }
```

Outro Programa

Escrevendo inteiros

```
1 #include <stdio.h>
2 int main(){
3     int idade=15;
4     printf("Maria tem %d anos.",idade);
5     return(0);
6 }
```


Constantes

- *Constantes* referem-se a valores fixos que o programa não pode alterar.
- Podem ser de qualquer tipo um dos cinco tipos de dados básicos.
- São definidas antes do início do programa.

Forma geral:

```
#define nome_da_constante valor_da_constante;
```

Exemplo

```
1 #define PI 3.14
```

Exemplo

Calculando área de uma circunferência

```
1 #include <stdio.h>
2 #define PI 3.14
3 int main(){
4     float raio=1.5;
5     float area = PI*raio*raio;
6     printf("A area da circunferencia e: %f",area);
7     return(0);
8 }
```

Operador de Atribuição

- Pode ser usado dentro de qualquer expressão válida de C.

Forma geral:

nome_da_variável = expressão;

- A *regra de conversão de tipos* diz que o valor do lado direito de uma atribuição é convertido no tipo do lado esquerdo.

Tipo do destino	Tipo da expressão	Possível perda
char	int	Os 8 bits mais significativos
int	float	A parte fracionária e possivelmente mais
float	double	Precisão, ocorre arredondamento

Operadores Aritméticos

Operador	Ação
-	Subtração
+	Adição
*	Multiplicação
/	Divisão
%	Módulo da divisão (resto)
--	Decremento
++	Incremento

A função scanf

- Realiza a leitura de um texto a partir do teclado.
- Parâmetros:
 - Uma string, indicando os tipos das variáveis que serão lidas e o formato dessa leitura.
 - Uma lista de variáveis.
- Aguarda que o usuário digite um valor e atribui o digitado à variável.

A função scanf

A função `scanf()` tem a seguinte forma geral:

```
scanf(string_de_controle, lista_de_argumentos);
```

String de controle

Código	Significado
<code>%d</code>	inteiro
<code>%f</code>	float
<code>%c</code>	caractere

A função scanf

Exemplo

```
1 #include <stdio.h>
2 int main(){
3     int n;
4     printf("Digite um número inteiro: ");
5     scanf("%d",&n);
6     printf("O valor digitado foi: %d",n);
7     return(0);
8 }
```


Leitura de várias variáveis

Exemplo

```
1 #include <stdio.h>
2 int main(){
3     int m,n,o;
4     printf("Digite três números inteiros: ");
5     scanf("%d %d %d",&m, &n, &o);
6     printf("Os valores digitados foram: %d %d %d",m, n, o);
7     return(0);
8 }
```

Exercícios

- 1 Faça um programa que realiza a leitura de dois números inteiros e imprima a soma entre eles.
- 2