

Expressões e Entrada e Saída de Dados

Prof. Thiago Oliveira dos Santos
Departamento de Informática
Universidade Federal do Espírito Santo

2015

Visão Geral da Aula



Informática

- Expressões
- Operadores
- Instruções primitivas

Conceito

- Intimamente ligado ao de expressões matemáticas
- Na matemática, é o relacionamento
 - Entre variáveis e constantes numéricas, e subexpressões
 - Através de operadores aritméticos
 - Compõe uma formula e produz um valor
- Exemplo
 - Formula de cálculo da área de um triângulo
 - $0.5 * B * H$ (1 constante, 2 variáveis e 2 operadores de mult.)
- Na computação, é mais amplo
 - O valor produzido não está restrito a números
 - Mas ainda formada por variáveis, constantes e operadores

Conceito

- Valor de um determinado tipo
- Não muda durante a execução do programa
- Exemplo
 - Inteira: 10, 0, 20, -1, -100,...
 - Real: 0.0, 10.0, .35, -1.0,...
 - Lógica: 0 (falso), -2 (verdadeiro), 1 (verdadeiro), ...
 - Caractere: 'a', 'g', '1', ...

Identificador de constante

- As vezes é necessário reutilizar um mesmo valor várias vezes
- Repetir sua escrita dificulta leitura e alteração do programa
- Nesses casos, usa-se a diretiva *#define*
 - Deve ser definida no início do programa
 - Sintaxe: *#define <identificador> <valor>*
 - Exemplo

```
#define PI 3.141593
```

```
int main() {  
    double raio, area, comprimento;  
    raio = 10;  
    area = PI * raio * raio;  
    comprimento = 2 * PI * raio;  
    return 0;  
}
```

Sugestão de padrão para identificador de constantes: Escrever tudo em maiúsculo separado por "_". Ex: ALTURA_MAXIMA

Operadores



Informática

Conceito

- Elementos funcionais que atuam sobre operandos
- Produzem um valor
- Exemplo
 - $2 + 3$
 - Relaciona dois operandos (2 e 3)
 - Realiza a operação de adição

Classificação

- De acordo com o número de operandos
 - Binário
 - Unário

Binário

- Atuam sobre dois elementos
- Exemplo: adição, subtração, multiplicação, divisão, etc.

Unário

- Atuam sobre um único elemento
- Sinal de (-) na frente de um número (faz inversão de sinal)

Classificação

- De acordo com o tipo de dados e o valor resultante
 - Aritmético, lógico e literal
- Portanto, diretamente ligado ao tipo de expressão

Exceção

- Operadores relacionais
 - Permite comparar 2 operandos de mesmo tipo
 - Retorna sempre um valor lógico

Tipo de Expressões



Informática

Classificação

- Tipos
 - Aritméticas
 - Relacionais
 - Lógicas

Expressões Aritméticas

Conceito

- Utilizada para fazer cálculos
- Resultado do tipo numérico (inteiro ou real)
- Componentes
 - Variáveis numéricas (tipo inteiro ou real)
 - Constantes numéricas
 - Operadores aritméticos
- Alguns operadores aritméticos em ordem de prioridade
 - 1) - (inversão de sinal), + (manutenção de sinal)
 - 2) * (multiplicação), / (divisão), % (resto da divisão inteira)
 - 3) + (adição), - (subtração)

Expressões Aritméticas

Prioridade de operadores

- Define a ordem em que os mesmos devem ser avaliados
- Pode ser mudada com o uso de parênteses
- Exemplo
 - $2 + 2 * -3$???

Tipo de retorno

- Inteiro, se todos os tipos são inteiros
- Real, se ao menos uma variável ou constante é real
- Exemplos (A, B, C são int e X, Y, Z são float)
 - $A + B * C$, resultado do tipo??
 - $A + B + Y$, resultado do tipo??
 - A / B , resultado do tipo?? **Importante! $3/2$ vai ser 1 e $1.0 + 3/2$ vai ser 2.0**
 - X / Y , resultado do tipo?? **Importante! $3.0/2$ vai ser 1.5**

Expressões Aritméticas



Informática

Outras Operações

- Outras funções matemáticas podem ser encontradas
 - Seno (sin), Coseno (cos), raiz quadrada (sqrt), etc.
 - Porém, é necessário a inclusão da biblioteca math.h
- Sintaxe geral: <nome_da_função>(<valor>)
- Exemplo

```
#include <math.h>

int main() {
    double a = 40, b = 60, raizDaSoma;

    raizDaSoma = sqrt(a + b);

    return 0;
}
```

Expressões Relacionais

Conceito

- Realiza comparações entre objetos de mesmo tipo
- Resulta em um valor lógico (0 para falso e 1 para verdadeiro)
- Operadores relacionais:
 - “>” Maior que
 - “>=” Maior que ou igual
 - “<” Menor que
 - “<=” Menor que ou igual
 - “==” Igual – **CAUIDADO! Não confundir com atribuição.**
 - “!=” Diferente
- Exemplo
 - $2 < 2$ retorna 0, $10 \leq 10$ retorna 1, $3 > 5$ retorna 0, etc.

Expressões Lógicas

Conceito

- Utilizadas para relacionar valores lógicos
- Muito usada para tradução literal das palavras “e” e “ou”
- Resulta em um valor lógico (0 para falso e 1 para verdadeiro)
- Operadores lógicos
 - “&&” E Cuidado! Esquecer um & ou um | resulta em outro operador. Erro difícil de mapear.
 - “||” Ou
 - “!” Não
- Exemplo
 - 2 && 2 retorna 1, 0 || 1 retorna 1, !0 retorna 1

Expressões Lógicas

Exemplo

- a, b com valores lógicos
- A e B podem assumir dois valores
 - Falso, 0
 - Verdadeiro, qualquer valor diferente de 0
- Possíveis combinações dos valores de a e b

a	b	$a \&\& b$	$a b$	$!a$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Ordem de Avaliação

- Operadores de maior prioridade são avaliados primeiro
- Operadores com mesma prioridade
 - Não estabelecida
 - Depende do compilador
 - Não confie! Se precisar garantir, use parênteses para garantir.
 - Exemplo:
 - $X = \text{expressão}_1 + \text{expressão}_2$
 - Quem vem primeiro expressão 1 ou 2?

Operadores



Informática

Resumo da Precedências

- Alguns operadores ainda não foram vistos

Primeiro	() [] -> .	
	- ++ -- ! & * ~ (type) sizeof	operadores unários
	* / %	
	+ -	
	<< >>	
	< <= >= >	
	== !=	
	&	
	^	
	&&	
	? :	
	= op=	
Último	,	

Avaliação de Expressões

Exemplos

- Considere que
 - X, Y, Z são reais e A e B são lógicas
 - $X=2.0$, $Y=3.0$, $Z=0.5$, $A=1$, $B=0$
- Avalie
 - $X * Y - Z$
 - $X * (Y - Z)$
 - $X + Y * Z$
 - $X + (Y * Z)$
 - $(X + Y) * Z$
 - $B \&\& A \mid\mid X \neq Y/2$
 - $B \&\&(A \mid\mid X \neq Y/2)$

Exemplos

- Considere que
 - X, Y, Z são reais e A e B são lógicas
 - $X=2.0$, $Y=3.0$, $Z=0.5$, $A=1$, $B=0$
- Avalie
 - $X * Y - Z = 5.5$
 - $X * (Y - Z) = 5.0$
 - $X + Y * Z = 3.5$
 - $X + (Y * Z) = 3.5$
 - $(X + Y) * Z = 2.5$
 - $B \ \&\& \ A \ || \ X \ != \ Y/2 = 1$
 - $B \ \&\&(A \ || \ X \ != \ Y/2) = 0$

Conceito

- Comandos básicos
- Efetua tarefas essenciais
 - Entrada e saída de dados
 - Comunicação com dispositivos de entrada e saída
- Presentes em “todas” linguagens de programação
- Definidas por palavras ou símbolos reservados
- Respeitam uma regra de sintaxe
- Possuem uma semântica
- Necessárias para a comunicação com o mundo exterior

Entrada e Saída de Dados



Informática

Dispositivos de Entrada

- Meio pelo qual dados são transferidos para memória principal
- Origem dos dados
 - Usuário (teclado, mouse, etc.)
 - Dispositivos secundários de memória (Disco rígido, USB, etc.)

Dispositivos de Saída

- Meio pelo qual dados são transferidos da memória principal
- Destino dos dados
 - Usuário (monitor, impressora, etc.)
 - Dispositivos secundários de memória (Disco rígido, USB, etc.)

Entrada e Saída de Dados

Entrada de Dados

- Sintaxe
 - `scanf ("<Formato_1><Formato_2>...", &var1, &var2,...);`
- Semântica
 - Espera enquanto o buffer estiver vazio, e lê entrada do teclado
- Requer inclusão da linha
 - `#include <stdio.h>`

Formato	Tipo
%d	Int
%f	Float
%c	Char

Entrada e Saída de Dados



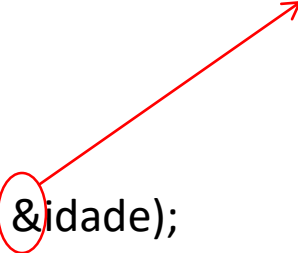
Informática

Entrada de Dados

- Exemplo

```
int main() {  
    int idade;  
    scanf("%d", &idade);  
    printf("Você tem %d anos.\n", idade);  
    return 0;  
}
```

Atentar para o "&"
Maior fonte de erros para iniciantes

A red arrow originates from the text "Atentar para o '&'" and points to the ampersand character in the scanf function call of the code example.

Entrada e Saída de Dados

Saída de Dados

- Sintaxe
 - `printf("<Formato_1><Formato_2>...", exp1, exp2,...);`
- Semântica
 - Imprime na tela do computador
- Requer inclusão da linha
 - `#include <stdio.h>`
- Caracteres especiais são precedidos por `\` e o símbolo `%`, por %
 - Exemplo
 - `\n` = quebra de linha, `\t` = tabulação, `%%` imprime %
 - Pesquise outros no manual

Formato	Tipo
%d	Int
%f	Float
%c	Char

Entrada e Saída de Dados



Informática

Saída de Dados

- Exemplos

Comando	Impresso na tela
<pre>printf("Oi\nComo vai? ");</pre>	Oi Como vai?
<pre>int x; float a; x = 10; a = 100.0; printf("x eh: %d\n", x); printf("a eh: %f\n", a); printf("2*a eh: %f\n2*x eh:%d\n", 2*a, 2*x);</pre>	x eh: 10 a eh: 100.000000 2*a eh: 200.000000 2*x eh:20

Implícita

Cuidado!

- Feita pelo compilador sem ordem direta do programador
- Transforma um tipo de dados em outro
- Regras da atribuição
 - Retorno da expressão é convertido para o tipo da variável
 - Exemplo

```
int x = 1000;  
char ch = 0;  
float f = 10.1;
```

```
ch = x; // Só aceita de 0 a 255. Bits mais significativos são ignorados  
x = f; // Recebe a parte inteira de f (10)  
f = ch; // Converte valor de ch para real  
f = x; // Converte inteiro para real. Pode perde precisão.
```

Conversão de Tipos

Explícita (Cast)

- Feita diretamente pelo programador
- Transforma um tipo de dados em outro
- Faça conversão explícita (cast)
- Sintaxe
 - (<tipo>) <expressão>
- Exemplo

```
int x = 1, z = 2;  
float y;
```

```
y = x / z; // y = 0  
y = (float)x / z; // y = 0.5  
y = x / (float)z; // y = 0.5  
y = (float) (x / z); // y = 0
```

Perguntas???



Informática



Exercícios



Informática

Escreva um programa para calcular o consumo médio de um automóvel (Km/l), dado que são conhecidos a distância total percorrida e o volume de combustível consumido para percorrê-la (medido em litros).

Exercícios



Informática

```
#include <stdio.h>

int main() {
    float totalDist, combCons, media;

    printf("Digite a distancia total:");
    scanf("%f", &totalDist);

    printf("Digite o combustivel consumido");
    scanf("%f", &combCons);

    media = totalDist / combCons;

    printf("Consumo medio: %f", media);
    return 0;
}
```