



Técnicas de Programação I
Prof. Dr. Antonio Marcos SELMINI
profselmini@uol.com.br
selmini@fiap.com.br

Introdução à Linguagem C

Introdução

- A linguagem de programação C surgiu em 1972;
- É uma linguagem de programação de uso geral desenvolvida por Dennis Ritchie no Bell Laboratory;
- Os princípios básicos de C são originários de outras linguagens tais como: B, BCPL e CPL;
- Estas linguagens são classificadas como linguagens de alto nível, mas que permitem a manipulação das informações através dos bits;

Introdução

- **Inicialmente a linguagem C ficou restrita as Universidades e aos centros de pesquisas;**
- **A maioria dos programadores achava que era uma linguagem de difícil manipulação;**
- **O surgimento da linguagem C está intimamente relacionada com o sistema operacional UNIX;**
- **Ken Thompson utilizou a linguagem Assembly e a linguagem B para produzir versões do sistema UNIX e a linguagem C surge devido as limitações da linguagem B;**

Tipos de dados primitivos em C

- A linguagem C apresenta basicamente cinco tipos de dados primitivos:
 - Caractere – char
 - Inteiro – int
 - Ponto flutuante (real) – float
 - Ponto flutuante de precisão dupla (real) – double
 - Sem valor – void

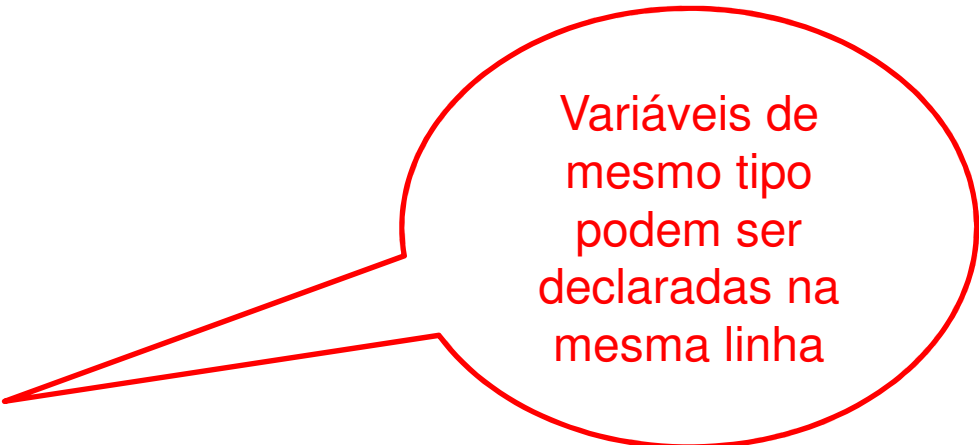
- A declaração de uma variável em C é feita da mesma forma que foi utilizada em pseudocódigo: primeiro o tipo da variável e logo depois o nome da variável;

Tipos de dados primitivos em C

- Toda linha de comando em C é terminada por ponto e vírgula (;);

- Exemplos:

- `int x;`
- `int a, b;`
- `float g;`
- `double v, s;`
- `char nome;`



Variáveis de
mesmo tipo
podem ser
declaradas na
mesma linha

Tipos de dados primitivos em C

Tipo	Tamanho em bits	Tamanho em bytes	Faixa
char	8	1	-128 a 127
unsigned char	8	1	0 a 256
signed char	8	1	-128 a 127
int	32	4	-2.147.483.648 a 2.147.483.647
unsigned int	32	4	0 a 4.294.967.295
signed int	32	4	mesmo que int
short int	16	2	-32.768 a 32.767
unsigned short int	16	2	0 a 65535
signed short int	16	2	mesmo que short int
long int	32	4	mesmo que int
unsigned long int	32	4	0 a 4.294.967.295
signed long int	32	4	mesmo que long int
float	32	4	$3,4 \cdot 10^{-38}$ a $3,4 \cdot 10^{38}$
double	64	8	$1,7 \cdot 10^{-308}$ a $1,7 \cdot 10^{308}$
long double	128	16	$3,4 \cdot 10^{-4932}$ a $1,7 \cdot 10^{4932}$

Caracteres especiais

■ Alguns caracteres especiais presentes na linguagem C:

- **%d ou %i:** valores do tipo inteiro (int)
- **%f:** valores do tipo real (float ou double)
- **%c:** valores do tipo caractere (char)
- **%s:** valores do tipo string
- **%ld:** valores do tipo long int
- **%u:** valores do tipo unsigned
- **%lu:** valores do tipo unsigned long int
- **%hd:** valores do tipo short int
- **%hu:** valores do tipo unsigned short int
- **\n:** mudança de linha
- **\t:** tabulação

Operadores em C

- **Operador de atribuição: =;**
- **Operadores aritméticos: +, -, *, /, % (resto da divisão inteira);**
- **Operadores relacionais: >, >=, <, <=, ==, !=;**
- **Operadores lógicos: && (E), || (OU), ! (NÃO);**

Comando de saída de dados

- **imprima(): `printf()` → biblioteca: `stdio.h`;**
- **Exemplos:**
 - **`printf("Esta é uma mensagem de exemplo");`**
 - **`printf("Tenham todos uma\n boa noite");`**
 - **`printf("Sua nota é %f", y);`**
 - **`printf("Os valores são %d e %d", x, y);`**
 - **`printf("o resultado de %d + %d = %d\n", x, y, x+y);`**

Comando de entrada de dados

- **leia(): `scanf()` → biblioteca: `stdio.h`;**

- **Exemplos:**
 - `scanf("%d", &x);`
 - `scanf("%f", &media);`
 - `scanf("%c", &letra);`
 - `scanf("%d %d %d", &x, &y, &z);`
 - `scanf("%d", x);` **//não está correto! Falta o &**

Estrutura de um programa em C

```
#include <...>
int main()
{
    //1ª Declaração de variáveis
    //2ª Entrada de dados
    //3ª Processamento
    //4ª Saída de dados
    return(0);
}
```

Exemplo 1

```
#include <stdio.h>
int main()
{
    int y;
    printf("digite um valor inteiro: ");
    scanf("%d", &y);
    printf("o valor digitado foi %d", y );
    return(0);
}
```

Exemplo 2

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    printf("Os alunos são %2d\n", 350);
    printf("Os alunos são %4d\n", 350);
    printf("Os alunos são %6d\n", 350);
    system("pause");
    return(0);
}
```

**A função system() é definida
na biblioteca stdlib.h**

Exemplo 4

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    float y = 2.56;
    printf("y = %f\n", y); //impressão padrão (6 casas decimais)
    printf("y = %.2f\n", y); //impressão com 2 casas decimais
    printf("y = %.1f\n", y); //impressão com 1 casa decimal
    printf("y = %.0f\n", y); //impressão sem casa decimal
    system("pause");
    return(0);
}
```

Exercícios

1. Escreva um algoritmo que calcule e imprima no vídeo o valor em graus Fahrenheit de uma temperatura expressa em graus Celsius dada pela expressão abaixo:

$$F = \frac{9C}{5} + 32$$

2. O sistema de avaliação de uma disciplina é composto por três provas. A primeira prova tem peso 2, a segunda tem peso 5 e a terceira peso 3. Escreva um algoritmo que calcule a média de um aluno na disciplina.

Exercícios

3. Escreva um algoritmo que leia um valor inteiro de três dígitos e mostre o valor do meio (valor armazenado na dezena). Se o valor de entrada for 123 deverá ser exibido 2. **Observação: suponha que o usuário irá digitar apenas valores com três dígitos, ou seja, valores entre 100 e 999.**
4. Escreva um algoritmo que leia um valor inteiro composto por três dígitos. Inverta o valor lido e o armazene em outra variável. Por exemplo: valor lido = 235 deverá ser armazenado em outra variável o valor 532.

Exercícios

5. **Uma revendedora de carros paga a seus funcionários vendedores, dois salários mínimos fixo (valor vigente), mais uma comissão fixa de R\$ 650,00 por carro vendido e mais 7,5% do valor total das vendas. Escreva um algoritmo que calcule o valor do salário de um vendedor.**
6. **Escreva um algoritmo que leia um número entre 0 e 60 e imprima o seu sucessor, sabendo que o sucessor de 60 é 0. Não pode ser utilizado nenhum comando de seleção ou repetição.**

Bibliografia

- **DEITEL, Paul; DEITEL, Harvey. C – Como Programar. 6ª edição. São Paulo: Pearson Prentice Hall, 2011.**
- **FORBELLONE, André Luiz Villar; EBERSPÄCHER, Henri Frederico. Lógica de Programação. 3ª edição. São Paulo: Pearson Prentice Hall, 2005.**