

Algoritmos e Programação de Computadores

Disciplina: 113475

Profa. Carla Denise Castanho

Universidade de Brasília - UnB
Instituto de Ciências Exatas - IE
Departamento de Ciência da Computação - CIC

2. ALGORITMOS COMPUTACIONAIS

PROGRAMAÇÃO EM LINGUAGEM C



Programação em Linguagem C

Estrutura Básica de um Programa em C

```
#include <bibliotecas>
int main () {
    <declaração de variáveis>
    <instruções>
}
```

Programação em Linguagem C

Estrutura Básica de um Programa em C

```
#include <stdio.h>
int main () {
    <declaração de variáveis>
    <instruções>
}
```

- ▶ Inclui a biblioteca **stdio** (*standard input/output*), utilizada para ler dados do teclado (com a função **scanf**) e escrever na tela (com a função **printf**).

Programação em Linguagem C

Estrutura Básica de um Programa em C

```
#include <stdio.h>
int main () { ←
    <declaração de variáveis>
    <instruções>
}
```

- ▶ Inclui a biblioteca **stdio** (*standard input/output*), utilizada para ler dados do teclado (com a função **scanf**) e escrever na tela (com a função **printf**).
- ▶ Declara uma função chamada **main**, o compilador sabe que uma função com este nome especial deve ser a **primeira função a ser executada** em um programa. Todo programa C deve ter uma função main, e ela deve ser única.

Programação em Linguagem C

Estrutura Básica de um Programa em C

```
#include <stdio.h>
int main () {
    <declaração de variáveis>
    <instruções>
}
```

- ▶ Inclui a biblioteca **stdio** (*standard input/output*), utilizada para ler dados do teclado (com a função **scanf**) e escrever na tela (com a função **printf**).
- ▶ Declara uma função chamada **main**, o compilador sabe que uma função com este nome especial deve ser a **primeira função a ser executada** em um programa. Todo programa C deve ter uma função main, e ela deve ser única.
- ▶ Delimitadores de início e de final de bloco (definem o corpo da função).

Linguagem C - Comentários

- Utilize comentários entre `/*` e `*/` para explicar melhor o que seu código faz!

Programa C com comentários

```
#include <stdio.h>
/* Isto eh um comentario, ele serah ignorado pelo compilador. Nao use
acentos ou caracteres especiais, pois nao sao C ANSI! */

/* Programa que calcula a media de dois numeros. */
int main () {
    float media; /* declara uma variavel chamada media */
    media = (4 + 5) / 2.0;

    /* fim do programa normal */
    return 0;
}
```

Linguagem C - Operadores

Operador	Operação	Linguagem C
=	Igual/Igualdade	==
≠	Diferente/Desigualdade	!=
<	Menor	<
≤	Menor ou Igual	<=
>	Maior	>
≥	Maior ou Igual	>=
OU	Disjunção	
E	Conjunção	&&
NÃO	Negação	!
×	Multiplicação	*
÷	Divisão	/
+	Adição	+
-	Subtração	-
Resto ou mod	Resto da Divisão	%
←	Atribuição	=

Linguagem C - Tipos de Variáveis

Tipo	Linguagem C	Bytes	Faixa de valores
Caractere, Inteiro	<code>char</code>	1	-128 a 127
Caractere, Inteiro	<code>unsigned char</code>	1	0 a 255
Caractere, Inteiro	<code>signed char</code>	1	-128 a 127
Inteiro	<code>int</code>	4	-2.147.483.648 a 2.147.483.647
Inteiro	<code>unsigned int</code>	4	0 a 4.294.967.295
Inteiro	<code>signed int</code>	4	-2.147.483.648 a 2.147.483.647
Inteiro	<code>short int</code>	2	-32.768 a 32.767
Inteiro	<code>unsigned short int</code>	2	0 a 65.535
Inteiro	<code>signed short int</code>	2	-32.768 a 32.767
Inteiro	<code>long int</code>	4	-2.147.483.648 a 2.147.483.647
Inteiro	<code>signed long int</code>	4	-2.147.483.648 a 2.147.483.647
Inteiro	<code>unsigned long int</code>	4	0 a 4.294.967.295
Real	<code>float</code>	4	$\pm 3.4028234e\pm 38$
Real	<code>double</code>	8	$\pm 1.79769313486223157e\pm 308$
Real	<code>long double</code>	10	Limites máximos da linguagem

Linguagem C - Tipos de Variáveis (resumo)

Tipo	Linguagem C
Inteiro	<code>int</code>
Real	<code>float</code>
Caractere	<code>char</code>
String	<code>char</code> nome[tamanho] ex: <code>char</code> endereco[30]
Lógico	Usar <code>int</code> , será “falso” quando for zero, e “verdadeiro” quando não for zero.

Declaração de Variáveis e Constantes

▶ Declaração de Variáveis:

```
<tipo> <nome 1>, ... , <nome n>;
```

▶ Exemplo:

```
int num, val;  
float media, p1, p2;  
char sexo, est_civ;
```

▶ Uma constante pode ser declarada ou definida da seguinte forma:

```
const <tipo> <nome> = <valor ou expr>;
```

▶ Exemplo:

```
const int VALOR= 10;  
const float TAXA = 1.5;  
const char LETRA = 'S';
```

Definição de Macros

- ▶ A diretiva **#define** associa um identificador a um valor. É utilizado para definir constantes, vincular uma determinada informação a um identificador, ou criar macros.

```
#define <nome_macro> <valor ou expressão>
```

- ▶ Exemplo:

```
#define FALSE 0  
#define TRUE 1  
#define MSG "Digite S para sair"
```

O comando *printf*

- ▶ O comando **printf** é similar ao “Escreva” do pseudocódigo, servindo para mostrar mensagens ao usuário, além do conteúdo de variáveis.
- ▶ Para mostrar uma mensagem constante:
 - ▶ Ex: `printf("Olá mundo!\n");`
 - ▶ O caractere `\n` faz o programa saltar uma linha ao final da frase.
- ▶ Para mostrar o conteúdo de variáveis, é necessária a inserção de **modificadores** na *string* do *printf*:
 - ▶ Ex: `printf("Idade do funcionário: %d\n", idade);`
- ▶ Note que, após a *string* de mensagem, há uma lista de variáveis a serem mostradas, na mesma **ordem** em que seus modificadores foram inseridos, e que o **tipo** da variável deve ser compatível com o tipo indicado pelo modificador, como veremos a seguir.

O comando *printf* - modificadores

Tipo	Modificador
<code>int</code>	<code>%d</code>
<code>float</code>	<code>%f</code>
<code>float</code> com <i>n</i> casas decimais	<code>%.nf</code> onde <i>n</i> é o número de casas decimais ex: <code>%.2f</code> para duas casas decimais
<code>char</code>	<code>%c</code>
<code>string</code>	<code>%s</code>

O comando scanf

- ▶ O comando **scanf** é um equivalente em C ao “Leia” do pseudocódigo.
- ▶ Ele armazena o que o usuário digitou em uma variável no programa.
- ▶ Segue a mesma lógica de modificadores que o *printf* (usando os mesmos modificadores), porém, exige que seja passado o **ENDEREÇO** da variável, ou seja, colocando um **&** antes do nome da variável.
- ▶ Exemplo:

```
printf("Digite sua idade:\n");  
scanf("%d", &idade);
```

- ▶ **Atenção!** No caso **string** (vetor de caracteres) **não é necessário utilizar & antes de seu nome!** Exemplo:

```
char nome[30];  
printf("Digite seu nome:\n");  
scanf("%s", nome); /* nao usa & */
```

Linguagem C - Programa de Exemplo

Programa C - Exemplo

```
#include <stdio.h>

int main () {
    float raio, area;
    const float PI = 3.14;

    printf("Digite o raio do círculo\n");
    scanf("%f", &raio);
    area = PI * raio * raio;
    printf("A area do circulo de raio %.2f vale %.2f.\n", raio,
area);
    return 0;
}
```


Linguagem C - Exercícios

1. Faça um programa em C que leia um número inteiro e mostre seu sucessor e antecessor.
2. Faça um programa em C que leia a idade da pessoa expressa em anos, meses e dias e mostre-a expressa apenas em dias. Assuma que o ano tem sempre 365 dias e o mês sempre 30 dias.
3. O custo ao consumidor de um carro novo é a soma do custo de fábrica com a porcentagem do distribuidor e dos impostos (aplicados ao custo de fábrica). Supondo que a porcentagem do distribuidor seja de 28% e os impostos de 45%, escreva um programa em C que leia o custo de fábrica e imprima o custo ao consumidor.
4. Faça um programa C que leia a idade de uma pessoa expressa em dias e mostre-a expressa em anos, meses e dias. Assuma que o ano tem sempre 365 dias e o mês sempre 30 dias.
5. Faça um programa em C que leia as 3 notas de um aluno e calcule a média final deste aluno. Considerar que a média é ponderada e que o peso das notas é: 1, 3 e 4, respectivamente.