

INE5231 Computação Científica I

Prof. A. G. Silva

28 de março de 2017

Conteúdo programático

- O computador - [3 horas-aula]
- Representação de algoritmos - [3 horas-aula]:
- Linguagens de programação estruturadas [3 horas-aula]
- **Introdução à programação em C [6 horas-aula]**
- Programas envolvendo processos de repetição e seleção [6 horas-aula]
- Variáveis estruturadas unidimensionais homogêneas [9 horas-aula]
- Variáveis estruturadas multidimensionais homogêneas [6 horas-aula]
- Variáveis estruturadas heterogêneas [6 horas-aula]
- Subdivisão de problemas e subprogramação [6 horas-aula]
- Programação utilizando uma linguagem de computação técnica numérica [6 horas-aula]

+ Roteiro

- Sistemas de tipos
- Variáveis e operadores
- Entrada e Saída

+ Palavras Reservadas

*auto, break, case, char, const, continue,
default, do, double, else, enum, extern, float,
for, goto, if, int, long, register, return,
short, signed, sizeof, static, struct, switch,
typedef, union, unsigned, void, volatile, while*

Obs.: C é case sensitive

+ Tipos de Dados

- São as formas que utilizamos para representar dados
- C possui 5 tipos básicos:
 - **char, int, float, double e void**
- E 4 modificadores básicos:
 - **signed, unsigned, long e short**
 - Os 4 podem ser aplicados ao **int**
 - **long** pode ser aplicado ao **double**
 - **signed** e **unsigned** aplicados ao **char**

+ Tipos de Dados

- **char**: Caracter: O valor armazenado é um caractere. Caracteres geralmente são armazenados em códigos **ASCII**
- **int**: Número inteiro é o tipo padrão e o tamanho do conjunto que pode ser representado normalmente depende da máquina em que o programa está rodando.
- **float**: Número em ponto flutuante de precisão simples. São conhecidos normalmente como números reais.
- **double**: Número em ponto flutuante de precisão dupla
- **void**: Este tipo serve para indicar que um resultado não tem um tipo definido. Uma das aplicações deste tipo em C é criar um tipo vazio que pode posteriormente ser modificado para um dos tipos anteriores.
- Mais informações:
 - <http://equipe.nce.ufrj.br/adriano/c/apostila/tipos.htm>

Tabela ASCII (American Standard Code for Information Interchange)

- <http://www.ime.usp.br/~hitoshi/introducao/20-caracteres.pdf>

032	!	033	"	034	#	035	\$	036	%	037	&	038	'	039	
(040)	041	*	042	+	043	,	044	-	045	.	046	/	047
0	048	1	049	2	050	3	051	4	052	5	053	6	054	7	055
8	056	9	057	:	058	;	059	<	060	=	061	>	062	?	063
@	064	A	065	B	066	C	067	D	068	E	069	F	070	G	071
H	072	I	073	J	074	K	075	L	076	M	077	N	078	O	079
P	080	Q	081	R	082	S	083	T	084	U	085	V	086	W	087
X	088	Y	089	Z	090	[091	\	092]	093	^	094	_	095
'	096	a	097	b	098	c	099	d	100	e	101	f	102	g	103
h	104	i	105	j	106	k	107	l	108	m	109	n	110	o	111
p	112	q	113	r	114	s	115	t	116	u	117	v	118	w	119
x	120	y	121	z	122	{	123		124	}	125	~	126		127

+ Variáveis

- Declaração:
 - *tipo nome = inicialização;*
- Escopo da variáveis:
 - globais: podem ser usadas em qualquer lugar do programa
 - locais: podem ser usadas apenas na função onde foi declarada

+ Exemplo

```
int a, b = 10; // Variáveis globais

void f(char c) {

    double d = 10.0; // Variável local

    int i = a; // Variável local

    // ...

}

int main() {

    int i = b; // Variável local

    return 0;

}
```

+ Operadores Aritméticos

Operador	Ação
+	Soma
-	Subtração ou troca de sinal
*	Multiplicação
/	Divisão
%	Resto da divisão inteira
++	Incremento
--	Decremento

+ Exercícios

- 1) Qual o valor das variáveis x, y e z após o seguinte trecho de código:

```
int x, y, z;  
x = y = 10;  
z = ++x;  
x = -x;  
y++;  
x = x + y - (z--);
```

- 2) Utilize o DEVCCPP para compilar e rodar código do exercício anterior.

+ Operadores Relacionais

Operador	Relação
>	Maior que
>=	Maior que ou igual a
<	Menor que
<=	Menor que ou igual a
==	Igual a
!=	Diferente de

+ Operadores/Conectivos Lógicos

Operador	Função
&&	AND
 	OR
!	NOT

+ Operadores Lógicos Bit a Bit

Operador	Ação
&	AND Lógico
 	OR Lógico
^	XOR (OR exclusivo)
~	NOT
>>	Shift Right
<<	Shift Left

+ Exercícios

- 3) Qual o valor das variáveis a, b, c, d, e, f após a execução do seguinte trecho de código:

```
int x = 2, y = 4;
int a, b, c, d, e, f;
a = x & y;
b = x | y;
c = x ^ y;
d = ~x;
e = x << 3;
f = x >> 1;
```


+ Casts

- Sintaxe:
 - *(tipo) expressão*

- Exemplo:

```
long a = 10, b = 4;
```

```
// f = 2.5
```

```
double f = (double) a/b;
```

+ Entrada e Saída de Dados

- A função `scanf` é utilizada para a leitura dados do dispositivo de entrada padrão
- A função `printf` é utilizada para a escrita de dados do dispositivo de saída padrão

+ Exemplos

```
int a;

scanf("%d", &a); // lê a

printf("%d", a); // escreve a

char ch;

scanf("%c", &ch); // lê c

printf("%c", ch); // escreve c

float num;

scanf("%f", &num); // lê f

printf("%f", num); // escreve f
```

Obs.: Cuidado para não esquecer do & no uso da função scanf

+ Tabela de Formato para E/S

Tipo	Formato para escrita e leitura
<code>[signed unsigned] char</code>	<code>%c</code>
<code>[signed] int</code>	<code>%i</code> ou <code>%d</code>
<code>unsigned int</code>	<code>%u</code>
<code>[signed] short int</code>	<code>%hi</code>
<code>unsigned short int</code>	<code>%hu</code>
<code>[signed] long int</code>	<code>%li</code>
<code>unsigned long int</code>	<code>%lu</code>
<code>float</code>	<code>%f</code>
<code>double</code>	<code>%lf</code>
<code>long double</code>	<code>%Lf</code>

+ Constantes de Barra Invertida

Código	Significado
<code>\b</code>	Retrocesso (backspace)
<code>\f</code>	Alimentação de Formulário (form feed)
<code>\t</code>	Tabulação Horizontal (tab)
<code>\n</code>	Nova Linha
<code>\"</code>	Aspas
<code>\'</code>	Apostrofo
<code>\0</code>	Nulo
<code>\\</code>	Barra Invertida
<code>\a</code>	Sinal Sonoro (Beep)
<code>\N</code>	Constante Octal (N é o valor da constante)
<code>\xN</code>	Constante Hexadecimal (N é o valor da constante)

+ Referências

- Allan Lima, Gustavo Henrique Porto, Pedro Silva Leite
 - <http://allanlima.wordpress.com/>
- Matos, P. A. & Carvalho, G. H. P. - A Linguagem de Programação C
- The C Programming Language
 - <http://www.engin.umd.umich.edu/CIS/course.des/cis400/c/c.html>
- Curso de C da UFMG
 - <http://ead1.eee.ufmg.br/cursos/C/>
- Lammert Bies, ASCII character map
 - <http://www.lammertbies.nl/comm/info/ascii-characters.html>
- Slides de Gustavo Henrique Porto do curso de C/C++

- Você pode:
 - copiar, distribuir, exibir e executar a obra
 - criar obras derivadas
 - fazer uso comercial da obra
- Sob as seguintes condições:
 - **Atribuição.** Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.
 - **Compartilhamento pela mesma Licença.** Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.
 - Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
 - Qualquer uma destas condições podem ser renunciadas, desde que Você obtenha permissão do autor.
- **Veja aqui a licença completa**

<https://creativecommons.org/licenses/by-sa/2.5/br/>