INSTITUTO FEDERAL
PARANÁ

LÓGICA DE PROGRAMAÇÃO

PROFª. M.Sc. JULIANA H Q BENACCHIO

Modificadores de Tipos

INSTITUTO FEDERAL
PARANÁ

- Os tipos de dados básicos em C podem estar acompanhados por modificadores na declaração de variáveis.
- Tais modificadores são: **`long`**, **`short`**, **`signed`** e **`unsigned`**.
- Os dois primeiros têm impacto no tamanho (número de bits) usados para representar um valor e os dois últimos indicam se o tipo será usado para representar valores negativos e positivos (**`signed`**) ou apenas positivos (**`unsigned`**).

Modificadores de Tipos

INSTITUTO FEDERAL
PARANÁ

Tipo	Significado	Tamanho (bytes)	Intervalo
char	Caractere	1	-128 a 127
signed char	Caractere com sinal	1	-128 a 127
unsigned char	Caractere sem sinal	1	0 a 255
int	Inteiro	2	-32.768 a 32.767
signed int	Inteiro com sinal	2	-32.768 a 32.767
unsigned int	Inteiro sem sinal	2	0 a 65.535
short int	Inteiro curto	2	-32.768 a 32.767
signed short int	Inteiro curto com sinal	2	-32.768 a 32.767
unsigned short int	Inteiro curto sem sinal	2	0 a 65.535
long int	Inteiro longo	4	-2.147.483.648 a 2.147.483.647
signed long int	Inteiro longo com sinal	4	-2.147.483.648 a 2.147.483.647
unsigned long int	Inteiro longo sem sinal	4	0 a 4.294.967.295

Modificadores de Tipos

INSTITUTO FEDERAL
PARANÁ

Tipo	Significado	Tamanho (bytes)	Intervalo
float	Ponto flutuante com precisão simples	4	$3.4*10^{-38}$ a $3.4*10^{38}$
double	Ponto flutuante com precisão dupla	8	$1.7*10^{-308}$ a $1.7*10^{308}$
long double	Ponto flutuante com precisão dupla longo	10	$3.4*10^{-4932}$ a $1.1*10^{4932}$

Inteiro (`int`)

INSTITUTO FEDERAL
PARANÁ

- Um número inteiro é um número sem vírgula, que pode ser expresso em diferentes bases:
- **Base decimal:** o número inteiro é representado por uma sequência de números unitários (de 0 a 9), que não deve começar por 0.
- **Base hexadecimal:** o número inteiro é representado por uma sequência de números unitários (de 0 a 9 ou de A a F (ou de a a f)), começando por 0x ou 0X.
- **Base octal:** o número inteiro é representado por uma sequência de números unitários (incluindo apenas números de 0 a 7), começando com 0.

Número Inteiro (`int`)

INSTITUTO FEDERAL
PARANÁ

- Os inteiros são assinados por padrão, o que significa que eles têm um sinal.
- Para armazenar informações sobre o sinal (em binário), os computadores usam o complemento de dois.

Ponto Flutuante (`float/double`)

- Um número com ponto flutuante é um número com vírgula, porém, ele pode ser representado de várias maneiras:
  - Um inteiro decimal: 895;
  - Um número com um ponto: 845.32;
  - Um número exponencial, ou seja, um número (eventualmente com vírgula) seguido da letra e (ou E) e de um inteiro correspondente à potência de 10 (assinado ou não, isto é, precedido de um + ou -):
    - 2.75e-2; 35.8E+10; .25e-2.

Ponto Flutuante (`float/double`)

INSTITUTO FEDERAL
PARANÁ

- Na verdade, os números reais são números com ponto flutuante, ou seja, um número em que a posição da vírgula não é fixa e é identificada por uma parte de seus bits (conhecido como o expoente), o restante dos bits permitem codificar o número sem vírgula (a mantissa).

Ponto Flutuante (`float/double`)

INSTITUTO FEDERAL
PARANÁ

- Os números do tipo **`float`** são codificados em 32 bits, incluindo:
  - 23 bits para a mantissa;
  - 8 bits para o expoente;
  - 1 bit para o sinal.

Ponto Flutuante (`float/double`)

INSTITUTO FEDERAL
PARANÁ

- Os números do tipo **`double`** são codificados em 64 bits, incluindo:
  - 52 bits para a mantissa;
  - 11 bits para o expoente;
  - 1 bit para o sinal.

Ponto Flutuante (`float/double`)

INSTITUTO FEDERAL
PARANÁ

- Os números do tipo **`long double`** são codificados em 80 bits, incluindo:
  - 64 bits para a mantissa;
  - 15 bits para o expoente;
  - 1 bit para o sinal.

Ponto Flutuante (`float/double`)

INSTITUTO FEDERAL
PARANÁ

- A precisão dos números reais é aproximada. Ela depende do número de posições decimais, dependendo do tipo real, ela será, no mínimo:
  - de 6 números para o tipo **`float`**;
  - de 15 números para o tipo **`double`**;
  - de 17 números para o tipo **`long double`**.

Caractere (`char`)

INSTITUTO FEDERAL
PARANÁ

- O tipo **`char`** (do inglês *character*) armazena o valor ASCII de um caractere, ou seja, um número inteiro.

ASCII control characters

00	NULL	(Null character)
01	SOH	(Start of Header)
02	STX	(Start of Text)
03	ETX	(End of Text)
04	EOT	(End of Trans.)
05	ENQ	(Enquiry)
06	ACK	(Acknowledgement)
07	BEL	(Bell)
08	BS	(Backspace)
09	HT	(Horizontal Tab)
10	LF	(Line feed)
11	VT	(Vertical Tab)
12	FF	(Form feed)
13	CR	(Carriage return)
14	SO	(Shift Out)
15	SI	(Shift In)
16	DLE	(Data link escape)
17	DC1	(Device control 1)
18	DC2	(Device control 2)
19	DC3	(Device control 3)
20	DC4	(Device control 4)
21	NAK	(Negative acknowl.)
22	SYN	(Synchronous idle)
23	ETB	(End of trans. block)
24	CAN	(Cancel)
25	EM	(End of medium)
26	SUB	(Substitute)
27	ESC	(Escape)
28	FS	(File separator)
29	GS	(Group separator)
30	RS	(Record separator)
31	US	(Unit separator)
127	DEL	(Delete)

ASCII printable characters

32	space	64	@	96	`
33	!	65	A	97	a
34	"	66	B	98	b
35	#	67	C	99	c
36	$	68	D	100	d
37	%	69	E	101	e
38	&	70	F	102	f
39	'	71	G	103	g
40	(	72	H	104	h
41	)	73	I	105	i
42	*	74	J	106	j
43	+	75	K	107	k
44	,	76	L	108	l
45	-	77	M	109	m
46	.	78	N	110	n
47	/	79	O	111	o
48	0	80	P	112	p
49	1	81	Q	113	q
50	2	82	R	114	r
51	3	83	S	115	s
52	4	84	T	116	t
53	5	85	U	117	u
54	6	86	V	118	v
55	7	87	W	119	w
56	8	88	X	120	x
57	9	89	Y	121	y
58	:	90	Z	122	z
59	;	91	[	123	{
60	<	92	\	124	|
61	=	93	]	125	}
62	>	94	^	126	~
63	?	95	_		

Extended ASCII characters

128	Ç	160	á	192	└	224	Ó
129	ü	161	í	193	┴	225	ß
130	é	162	ó	194	┬	226	Ô
131	â	163	ú	195	├	227	Ò
132	ä	164	ñ	196	─	228	õ
133	à	165	Ñ	197	┼	229	Õ
134	å	166	ª	198	ã	230	µ
135	ç	167	º	199	Ã	231	þ
136	ê	168	¿	200	╚	232	Þ
137	ë	169	®	201	╔	233	Ú
138	è	170	¬	202	╩	234	Û
139	ï	171	½	203	╦	235	Ù
140	î	172	¼	204	╠	236	ý
141	ì	173	¡	205	═	237	Ý
142	Ä	174	«	206	╬	238	¯
143	Å	175	»	207	¤	239	´
144	É	176	░	208	ð	240	≡
145	æ	177	▒	209	Ð	241	±
146	Æ	178	▓	210	Ê	242	‗
147	ô	179	│	211	Ë	243	¾
148	ö	180	┤	212	È	244	¶
149	ò	181	Á	213	ı	245	§
150	û	182	Â	214	Í	246	÷
151	ù	183	À	215	Î	247	¸
152	ÿ	184	©	216	Ï	248	°
153	Ö	185	╣	217	┘	249	¨
154	Ü	186	║	218	┌	250	·
155	ø	187	╗	219	█	251	¹
156	£	188	╝	220	▄	252	³
157	Ø	189	¢	221	¦	253	²
158	×	190	¥	222	Ì	254	■
159	ƒ	191	┐	223	▀	255	nbsp

Caractere (`char`)

INSTITUTO FEDERAL
PARANÁ

- Por padrão, os números são assinados, isso significa que eles têm um sinal. Para armazenar informações sobre o sinal (em binário), os computadores usam o complemento de dois.
- Então, um dado de tipo **`char`** é assinado, o que não significa que a letra possui um sinal, mas simplesmente que, na memória, o valor que codifica o caractere pode ser negativo.
- Se, por exemplo, quisermos armazenar a letra B (seu código ASCII é 66), poderemos definir este dado, seja pelo número 66, seja notando o 'B', onde as aspas simples significam **código ASCII de...**

Especificadores de formato

INSTITUTO FEDERAL
PARANÁ

Tipo de Dados	`printf / scanf`
char	`%c`
int	`%d`
unsigned int	`%u`
short	`%hd`
unsigned short	`%hu`
long int	`%ld`
unsigned long int	`%lu`
float	`%f`
double	`%f / %lf`
long double	`%Lf`

Especificadores de formato

INSTITUTO FEDERAL
PARANÁ

Saída	printf	Exemplo
Base Octal	`%o`	`610`
Base Hexadecimal	`%x`	`7fa`
Base Hexadecimal (maiúsculas)	`%X`	`7FA`
Notação Científica (exponencial)	`%e`	`3.9265e+2`
Notação Científica (exponencial maiúsculas)	`%e`	`3.9265E+2`
Geral, escolhe a mais curta entre `%f` e `%e`	`%g`	`392.65`
Hexadecimal ponto flutuante	`%a`	`-0xc.90fep-2`
Hexadecimal ponto flutuante (maiúsculas)	`%A`	`-0XC.90FEP-2`
String	`%s`	`texto`
Endereço (ponteiro)	`%p`	`b8000000`
Caractere %	`%%`	`10%`

Sequências de escape

INSTITUTO FEDERAL
PARANÁ

Sequência de escape	Descrição
`\n`	Nova linha. Posiciona o cursor da tela no início da próxima linha
`\t`	Tabulação horizontal. Move o cursor da tela para a próxima posição de tabulação
`\a`	Alerta. Faz soar o alarme do sistema
`\\`	Barra invertida. Insere um caractere de barra invertida em uma string
`\"`	Aspas. Insere um caractere de aspas em uma string

Formatos de impressão (`printf`)

- Por exemplo, podemos especificar o tamanho do número utilizado para impressão da seguinte forma:
- `%6d` → inteiro, com pelo tamanho pelo menos 6
- `%6f` → ponto flutuante, com tamanho pelo menos 6
- `%.3f` → ponto flutuante, com 3 dígitos depois do ponto decimal
- `%6.3f` → ponto flutuante, com tamanho pelo menos 6 e 3 dígitos depois do ponto decimal
- `%6.0f` → ponto flutuante, com pelo menos tamanho 6 e nenhum dígito depois do ponto decimal.