

类型、运算符与表达式

胡船长

初航我带你，远航靠自己

本期内容

- 一. 类型与变量
- 二. 输入&输出函数
- 三. 运算符与表达式
- 四. 课后实战题

一. 类型与变量

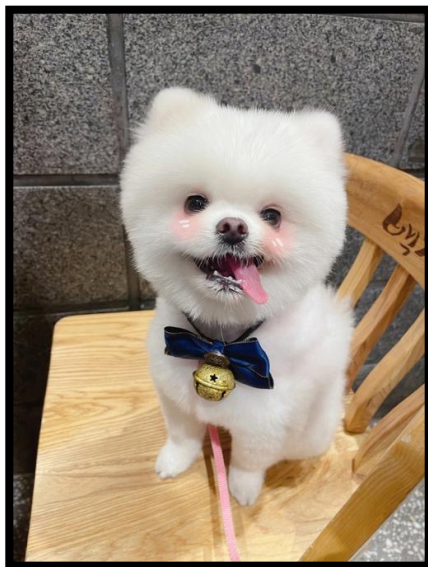
聊聊：类型与变量

聊聊：类型与变量

1. 来福是条狗
2. 德芙是只猫
3. 张三是一个人
4. A 是一个整型变量

聊聊：类型与变量

1. 来福是条狗
2. 德芙是只猫
3. 张三是一个人
4. A 是一个整型变量



聊聊：类型与变量

1. 来福是条狗
2. 德芙是只猫
3. 张三是一个人
4. A 是一个整型变量



聊聊：类型与变量

1. **来福**是条**狗**
2. **德芙**是只**猫**
3. **张三**是**个人**
4. A 是一个**整型变量**



聊聊：类型与变量

1. 来福是条狗
2. 德芙是只猫
3. 张三是一个人
4. A 是一个整型变量



聊聊：计算机中的数据存储

聊聊： 计算机中的数据存储

计算机存储数据的基本单位：字节



聊聊： 计算机中的数据存储

计算机**存储数据**的基本单位：字节

计算机**表示数据**的基本单位：位



聊聊： 计算机中的数据存储

计算机**存储数据**的基本单位：字节

计算机**表示数据**的基本单位：位

表示数据方式：二进制

0	1	0	0	1	1	0	1
---	---	---	---	---	---	---	---

聊聊： 计算机中的数据存储

计算机**存储数据**的基本单位：字节

计算机**表示数据**的基本单位：位

表示数据方式：二进制

0	1	0	0	1	1	0	1
---	---	---	---	---	---	---	---

					10^2	10^1	10^0
					5	3	1

聊聊： 计算机中的数据存储

计算机**存储数据**的基本单位：字节

计算机**表示数据**的基本单位：位

表示数据方式：二进制

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	1	0	0	1	1	0	1

					10^2	10^1	10^0
					5	3	1

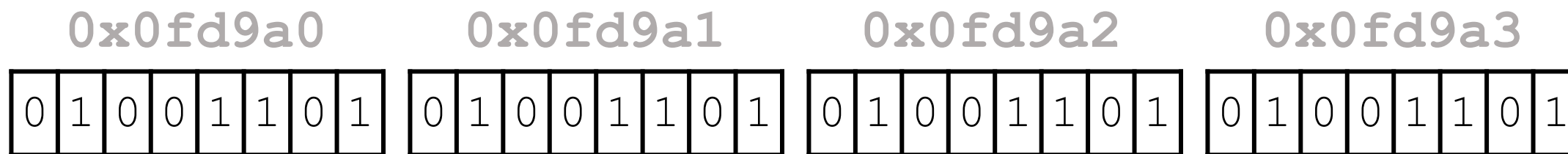
聊聊： 计算机中的数据存储

计算机**存储数据**的基本单位：字节

计算机**表示数据**的基本单位：位

表示数据方式：二进制

整型数据占**4**字节



聊聊：常用类型

聊聊：常用类型

123

97

100062

9651

聊聊：常用类型



聊聊：常用类型

整型：int | 格式占位符：%d

大小：4字节，32位

范围： -2^{31} (-2147483648) ~ $2^{31}-1$ (2147483647)



聊聊：常用类型

整型： `int` | 格式占位符： `%d`

大小： 4 字节， 32 位

范围： -2^{31} (-2147483648) \sim $2^{31}-1$ (2147483647)

聊聊：常用类型

长整型：long long | 格式占位符：%lld

大小：**8**字节，**64**位

范围： $-2^{63} \sim 2^{63}-1$



聊聊：常用类型



聊聊：常用类型

单精度浮点型：float | 格式占位符：%f

大小：4字节，32位

有效数字：7位

a

12.3

b

9.7

c

1000.62

d

9.651

聊聊：常用类型

单精度浮点型：float | 格式占位符：%f

大小：4字节，32位

有效数字：7位

a

12.3

b

9.7

c

1000.62

d

9.651

e

0.000123

聊聊：常用类型

双精度浮点型：double | 格式占位符：%lf

大小：8字节，64位

有效数字：15位

a

12.3

b

9.7

c

1000.62

d

9.651

e

0.000123

聊聊：常用类型

字符型：char | 格式占位符：%c

大小：1字节，8位

编码方式：ASCII编码

a

'c'

b

','

c

.'

d

'p'

e

'\$'

ASCII 编码对照表

二进制	十进制	十六进制	图形	二进制	十进制	十六进制	图形	二进制	十进制	十六进制	图形
0010 0000	32	20	(space)	0100 0000	64	40	@	0110 0000	96	60	`
0010 0001	33	21	!	0100 0001	65	41	A	0110 0001	97	61	a
0010 0010	34	22	"	0100 0010	66	42	B	0110 0010	98	62	b
0010 0011	35	23	#	0100 0011	67	43	C	0110 0011	99	63	c
0010 0100	36	24	\$	0100 0100	68	44	D	0110 0100	100	64	d
0010 0101	37	25	%	0100 0101	69	45	E	0110 0101	101	65	e
0010 0110	38	26	&	0100 0110	70	46	F	0110 0110	102	66	f
0010 0111	39	27	'	0100 0111	71	47	G	0110 0111	103	67	g
0010 1000	40	28	(0100 1000	72	48	H	0110 1000	104	68	h
0010 1001	41	29)	0100 1001	73	49	I	0110 1001	105	69	i
0010 1010	42	2A	*	0100 1010	74	4A	J	0110 1010	106	6A	j
0010 1011	43	2B	+	0100 1011	75	4B	K	0110 1011	107	6B	k
0010 1100	44	2C	,	0100 1100	76	4C	L	0110 1100	108	6C	l
0010 1101	45	2D	-	0100 1101	77	4D	M	0110 1101	109	6D	m
0010 1110	46	2E	.	0100 1110	78	4E	N	0110 1110	110	6E	n
0010 1111	47	2F	/	0100 1111	79	4F	O	0110 1111	111	6F	o
0011 0000	48	30	0	0101 0000	80	50	P	0111 0000	112	70	p
0011 0001	49	31	1	0101 0001	81	51	Q	0111 0001	113	71	q
0011 0010	50	32	2	0101 0010	82	52	R	0111 0010	114	72	r
0011 0011	51	33	3	0101 0011	83	53	S	0111 0011	115	73	s
0011 0100	52	34	4	0101 0100	84	54	T	0111 0100	116	74	t
0011 0101	53	35	5	0101 0101	85	55	U	0111 0101	117	75	u
0011 0110	54	36	6	0101 0110	86	56	V	0111 0110	118	76	v
0011 0111	55	37	7	0101 0111	87	57	W	0111 0111	119	77	w
0011 1000	56	38	8	0101 1000	88	58	X	0111 1000	120	78	x
0011 1001	57	39	9	0101 1001	89	59	Y	0111 1001	121	79	y
0011 1010	58	3A	:	0101 1010	90	5A	Z	0111 1010	122	7A	z
0011 1011	59	3B	;	0101 1011	91	5B	[0111 1011	123	7B	{
0011 1100	60	3C	<	0101 1100	92	5C	\	0111 1100	124	7C	
0011 1101	61	3D	=	0101 1101	93	5D]	0111 1101	125	7D	}
0011 1110	62	3E	>	0101 1110	94	5E	^	0111 1110	126	7E	~
0011 1111	63	3F	?	0101 1111	95	5F	_				

ASCII 编码对照表

二进制	十进制	十六进制	图形	二进制	十进制	十六进制	图形	二进制	十进制	十六进制	图形
0010 0000	32	20	(space)	0100 0000	64	40	@	0110 0000	96	60	`
0010 0001	33	21	!	0100 0001	65	41	A	0110 0001	97	61	a
0010 0010	34	22	"	0100 0010	66	42	B	0110 0010	98	62	b
0010 0011	35	23	#	0100 0011	67	43	C	0110 0011	99	63	c
0010 0100	36	24	\$	0100 0100	68	44	D	0110 0100	100	64	d
0010 0101	37	25	%	0100 0101	69	45	E	0110 0101	101	65	e
0010 0110	38	26	&	0100 0110	70	46	F	0110 0110	102	66	f
0010 0111	39	27	'	0100 0111	71	47	G	0110 0111	103	67	g
0010 1000	40	28	(0100 1000	72	48	H	0110 1000	104	68	h
0010 1001	41	29)	0100 1001	73	49	I	0110 1001	105	69	i
0010 1010	42	2A	*	0100 1010	74	4A	J	0110 1010	106	6A	j
0010 1011	43	2B	+	0100 1011	75	4B	K	0110 1011	107	6B	k
0010 1100	44	2C	,	0100 1100	76	4C	L	0110 1100	108	6C	l

ASCII 编码对照表

0010 1101	45	2D	-	0100 1101	77	4D	M	0110 1101	109	6D	m
0010 1110	46	2E	.	0100 1110	78	4E	N	0110 1110	110	6E	n
0010 1111	47	2F	/	0100 1111	79	4F	O	0110 1111	111	6F	o
0011 0000	48	30	0	0101 0000	80	50	P	0111 0000	112	70	p
0011 0001	49	31	1	0101 0001	81	51	Q	0111 0001	113	71	q
0011 0010	50	32	2	0101 0010	82	52	R	0111 0010	114	72	r
0011 0011	51	33	3	0101 0011	83	53	S	0111 0011	115	73	s
0011 0100	52	34	4	0101 0100	84	54	T	0111 0100	116	74	t
0011 0101	53	35	5	0101 0101	85	55	U	0111 0101	117	75	u
0011 0110	54	36	6	0101 0110	86	56	V	0111 0110	118	76	v
0011 0111	55	37	7	0101 0111	87	57	W	0111 0111	119	77	w
0011 1000	56	38	8	0101 1000	88	58	X	0111 1000	120	78	x
0011 1001	57	39	9	0101 1001	89	59	Y	0111 1001	121	79	y
0011 1010	58	3A	:	0101 1010	90	5A	Z	0111 1010	122	7A	z
0011 1011	59	3B	;	0101 1011	91	5B	[0111 1011	123	7B	{
0011 1100	60	3C	<	0101 1100	92	5C	\	0111 1100	124	7C	
0011 1101	61	3D	=	0101 1101	93	5D]	0111 1101	125	7D	}

一个你必须经常用到的网站

zh.cppreference.com

C 参考手册

C89, C95, C99, C11, C17, C23

语言

基本概念
关键词
预处理器
表达式
声明
初始化
函数
语句

头文件

类型支持

程序工具
变参数函数
错误处理
动态内存管理
字符串库

以空字符结尾的字符串：
字节 — 多字节 — 宽

算法

数值

常用数学函数
浮点环境 (C99)
伪随机数生成
复数算术 (C99)
泛型数学 (C99)

日期和时间工具

输入/输出支持

本地化支持

并发支持库 (C11)

技术规范

动态内存扩展 (动态内存 TR)

浮点扩展, 部分 1 (FP 扩展1 TS)

浮点扩展, 部分 4 (FP 扩展4 TS)

外部链接 — 非 ANSI/ISO 库 — 索引 — 符号索引

二. 输入&输出函数

输出函数说明

printf 函数

头文件：stdio.h

原型：int **printf**(const char *format, ...);

format：格式控制**字符串**

...：可变参数列表

返回值：**输出字符的数量**

输入函数说明

scanf 函数

头文件：stdio.h

原型：int **scanf**(const char *format, ...);

format：格式控制**字符串**

...：可变参数列表

返回值：**成功读入的参数个数**

随堂练习题-1

请使用 `printf` 函数，求解一个数字 `n` 的十进制表示的数字位数

随堂练习题-2

请写一个程序，读入一个行字符串（可能包含空格），输出这个字符串中字符的数量。

聊聊：sscanf & sprintf

随堂练习题-3

给输出的内容加一个漂亮的框框

Hello world

三. 运算符与表达式

C 语言基本运算符

运算符	说明	例子
=	赋值运算符	a = b;
+, -, *, /, ()	基本四则运算	a = (b + c) * d;
%	求余运算符	a = b % 2;
&, , ^, ~	位运算（非常重要的一类）	a = ~b c;
<<, >>	左移和右移	a = b >> 2;

运算符的优先级

优先级	运算符	描述	结合性
1	++ --	后缀自增与自减	从左到右
	()	函数调用	
	[]	数组下标	
	.	结构体与联合体成员访问	
	->	结构体与联合体成员通过指针访问	
	(type){list}	复合字面量(C99)	
2	++ --	前缀自增与自减 ^[注 1]	从右到左
	+ -	一元加与减	
	! ~	逻辑非与逐位非	
	(type)	转型	
	*	间接（解引用）	
	&	取址	
	sizeof _Alignof	取大小 ^[注 2] 对齐要求(C11)	
3	* / %	乘法、除法及余数	从左到右
4	+ -	加法及减法	
5	<< >>	逐位左移及右移	
6	< <=	分别为 < 与 ≤ 的关系运算符	
	> >=	分别为 > 与 ≥ 的关系运算符	
7	== !=	分别为 = 与 ≠ 关系	
8	&	逐位与	
9	^	逐位异或（排除或）	
10		逐位或（包含或）	
11	&&	逻辑与	
12		逻辑或	
13	?:	三元条件 ^[注 3]	从右到左
14 ^[注 4]	=	简单赋值	
	+= -=	以和及差赋值	
	*= /= %=	以积、商及余数赋值	
	<<= >>=	以逐位左移及右移赋值	
	&= ^= =	以逐位与、异或及或赋值	
15	,	逗号	从左到右

运算符的优先级

	_Alignof	对齐要求(C11)	
3	* / %	乘法、除法及余数	从左到右
4	+ -	加法及减法	
5	<< >>	逐位左移及右移	
6	< <= > >=	分别为 < 与 ≤ 的关系运算符 分别为 > 与 ≥ 的关系运算符	
7	== !=	分别为 = 与 ≠ 关系	
8	&	逐位与	
9	^	逐位异或（排除或）	
10		逐位或（包含或）	
11	&&	逻辑与	
12		逻辑或	
13	? :	三元条件[注 3]	从右到左
14[注 4]	= += -= *= /= %= <<= >>= &= ^= =	简单赋值 以和及差赋值 以积、商及余数赋值 以逐位左移及右移赋值 以逐位与、异或及或赋值	
15	,	逗号	从左到右

C 语言中的数学函数库

头文件: `math.h`

常用函数	常用函数
<code>pow(a, n)</code>	<code>fabs(n)</code>
<code>sqrt(n)</code>	<code>log(n)</code>
<code>ceil(n)</code>	<code>log10(n)</code>
<code>floor(n)</code>	<code>acos(n)</code>
<code>abs(n) (stdlib.h)</code>

pow-函数说明

pow 函数：指数函数

头文件：`math.h`

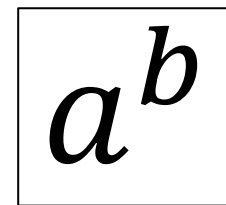
原型：`double pow(double a, double b);`

a：底数

b：指数

返回值：返回 a^b 的结果

例子：`pow(2, 3) = 8`


$$a^b$$

sqrt-函数说明

sqrt 函数：开平方函数

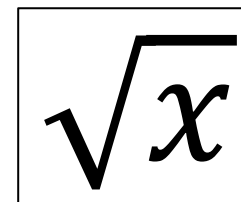
头文件：`math.h`

原型：`double sqrt(double x);`

`x`：被开方数

返回值：返回 \sqrt{x} 的结果

例子：`sqrt(16) = 4`



ceil-函数说明

ceil 函数：上取整函数

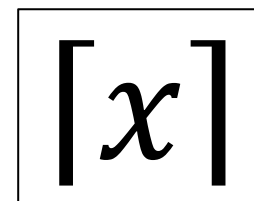
头文件：`math.h`

原型：`double ceil(double x);`

`x`：某个实数

返回值：返回 $\lceil x \rceil$ 的结果

例子：`ceil(4.1) = 5`



floor-函数说明

floor 函数：下取整函数

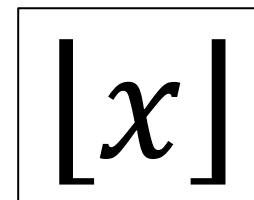
头文件：`math.h`

原型：`double floor(double x);`

`x`：某个实数

返回值：返回 $\lfloor x \rfloor$ 的结果

例子：`floor(4.9) = 4`



abs-函数说明

abs 函数：整数绝对值函数

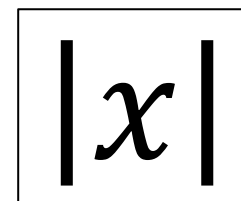
头文件：`stdlib.h`

原型：`int abs(int x);`

`x`：某个整数

返回值：返回 $|x|$ 的结果

例子：`abs(-4) = 4`



fabs-函数说明

fabs 函数：实数绝对值函数

头文件：`math.h`

原型：`double fabs (double x);`

`x`：某个实数

返回值：返回 $|x|$ 的结果

例子：`fabs (-4.5) = 4.5`

$$|x|$$

log-函数说明

log 函数：以 e 为底对数函数

头文件：`math.h`

原型：`double log(double x);`

x：某个实数

返回值：返回 $\log_e x$ 的结果

例子： $\log(9) = 2.197225\dots$

$$\log_e x$$

log10-函数说明

log10 函数：以 10 为底对数函数

头文件：`math.h`

原型：`double log10(double x);`

`x`：某个实数

返回值：返回 $\log_{10} x$ 的结果

例子：`log10(1000) = 3`

$$\log_{10} x$$

acos-函数说明

acos 函数

头文件：`math.h`

原型：`double acos(double x);`

x：角度的余弦值

返回值：以弧度值返回 `arccos(x)` 的结果

例子：`acos(-1) = 3.1415926...`

`arccos(x)`

随堂练习题-4

请写一个程序，输入一个数字 x ，输出数字 x 的立方根

随堂练习题-5

请写一个程序，读入一个角度值，将角度值转为弧度值。

四. 课后实战题

课后实战题

1-HZOJ-86: 三数的乘积

2-HZOJ-87: 矩形的面积与周长

3-HZOJ-90: 取数位 II

4-HZOJ-95: 交换两位数

5-HZOJ-97: 求和2

6-HZOJ-101: 计算各位和

7-HZOJ-102: 注水问题

不要考虑太多，坚持看完，
你就已经超过了95%的人。

5. 整型数据类型



| 3.58万次播放

54. 主函数参数



| 2892次播放