



一起++ 第2弹

变量与常量

作者：@孤言 审校：@Alex Cui

在第 1 弹中，我们对 C++ 的开发环境进行了配置。本弹将系统介绍变量。
这一弹内容较多，可以分段学习，每个知识点熟练掌握后，再进入下一个知识点。

知识点 1 注释

这部分内容作者之前疏忽忘写了，故而补之，敬请谅解/(T o T)/~~。

注释是对代码的解释和说明，其目的是提高代码的可读性。注释的书写应言简意赅。

```
1  /* HelloWorld.cpp - The first program
2     By GuYan */
3  #include <iostream>
4  using namespace std;
5  int main()
6  {
7      cout << "Hello World"; // 输出 Hello World
8      return 0;
9  }
```

C++ 中常有两种注释的书写方式。

书写单行注释时，要在该行注释开始处加上 `//`，其后书写注释，在行末注释结束。

书写多行注释时，要在该行注释开始处加上 `/*`，其后书写注释，注释结束处加上 `*/`。

与 Scratch 相同，注释不会影响程序执行。C++ 的注释在**编译**阶段即被忽略。虽然如此，写代码时，尤其是写大段代码时，仍应养成书写注释的好习惯。

知识点 2 数据类型

与某些编程语言“多个愿望，一次满足”的现象不同，C++ 的变量在声明之前需要明确其类型。不同的数据有适合它的不同类型。不同类型**一般**会有不同的宽度，即所能表示的范围是有限的。

我们先来了解一下在 C++ 中有哪些内置的数据类型。

1. 整型

整数

整数就是没有小数部分的数字，比如 -8，0，2020 等。

C++ 提供了多种整型，为了降低入门门槛，我们只认识其中常见的几种。

(1) int 型、long long 型

这两种类型一般用于表示整数。

在表示整数时，如无特殊情况，我们**首选 int 型**（高效），它的取值范围取决于环境，本教程采用的范围为 $-2^{31} \sim 2^{31}-1$ （下述的取值范围也仅是本教程采用的范围）

当数据范围超过 int 时，我们常会用到 long long 型，取值范围为 $-2^{63} \sim 2^{63}-1$ 。

(2) char 型

char 型是一种“特殊”的整型，通常用来表示字符。（为什么这也是整型呢？有基础的同学可参考篮框拓展）

在 C++ 中，我们将字符用 `' '` 括起来表示。像 `'@'`，`'A'`，`'5'` 等都是字符。当然，还存在空格 `' '`、换行符等字符。

char 型就是常用来表示这些基本字符的一种数据类型。

ASCII 码与 char 型

char 之所以属于整型，是因为这些字符与 **ASCII 码**（特殊情况下也可能是其它编码，但原理是一样的）中的数字编码对应，所以计算机真正存的不是字符本身，而是它对应的编码。

ASCII 码，又称**美国信息交换标准代码**，在扩展之前，一共收录了 128 个字符，下表展现了其中一部分。

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

当然，这些字符的编码是不需要特别记忆的，但最好记住常见的几个：数字 0 编码 48，大写字母 A 编码 65，小写字母 a 编码 97。其它的很多字母和数字的编码就可以推出来了。同一字母的大小写对应编码差 32。

(3) bool 型

相信你在 Scratch 中创建自定义积木的时候就已经发现了“布尔”这一术语。bool 型存放的是真 (`true`) 或假 (`false`)。

值得注意的是，在 C++ 中，数字 0 对应的 bool 型是 `false`，除 0 以外的数字为 `true`。

2.浮点型

浮点型通常用于表示实数，其中比较常见的是带有小数部分的数字（-123.4567，5.20 等）。

常见的浮点型有 float（**单精度浮点型**）和 double（**双精度浮点型**），它们在精度上存在着差异，通常选用精度较高的 double 型。

强制类型转换

C++ 程序执行时会隐式地对某些数据进行类型转换，但有时候这不是我们预期的效果，此时可以进行**强制类型转换**，常用格式为：

数据类型(数据)

注意 类型转换的结果是一个表达式。

比如在进行两个整型数据的除法时，如果想得到浮点型结果，可以这样写表达式：

```
double(5)/4
```

当然，也可以像下面这样写，但这就并不是强制类型转换了：

```
1.0*5/4
```

知识点 3 变量

1.变量的命名规则

每个人都有自己的姓名，比如法外狂徒张三，亦或友好公民李四。取名字往往还比较讲究，比如不会无缘无故起个王*蛋，也不会草草了事叫个孙悟空（万一呢）。

C++中对于变量的命名也是有规则的。

```
GuYan    alex_cui    g1024    _abc
52fjw    ^_^    return    变量 1
```

上面的第一行列出了一些合法的变量名。而**第二行为错误示例**。该规则具体阐述为：

- (1) 变量名只能由字母（A-Z，a-z）和数字（0-9）或者下划线（_）组成。
- (2) 必须是字母或者下划线开头，而不是数字开头。
- (3) 不能使用 C++保留字来命名变量，以免冲突。（保留字请自行查表，在编辑器中会用不一样的字体格式标注出来）
- (4) 变量名区分大小写（这一点对于没学过其它语言的同学来说不会奇怪）。

所以第二行的四个变量名错因依次是以数字开头、包含非法字符、使用了保留字、包含非法字符（中文字符）。

注意 变量名不能包含空格（违反第一条）。

以上规则不必刻意记忆，熟悉以后自然能够运用。

2.变量的声明

在 Scratch 中，使用变量前，我们需要建立一个变量。类似地，在 C++中使用变量，我们也需要对变量进行“创建”和命名，这一过程称为变量的**声明**。它的格式为：

```
数据类型 变量名；
```

以下是一些示例：

```
int a;
double num;
bool t;
```

这样，系统就对你所声明的变量分配了一块内存，这块内存的“代号”就是变量名。对于同种类型的变量，还可以一次性声明多个，用逗号隔开，如：

```
int a, b, c;
```

声明语句的书写位置不同可能会导致变量的作用范围不同，本弹不作探讨。

3.简单的变量赋值

我们可以用**=**来给变量赋值。格式为：

```
变量名 = 值；
```

例如：

```
int a;    //声明 int 型变量 a
```

建立一个变量

图 1

```
a = 5; //将变量 a 的值赋为 5
char ch; //声明 char 型变量 ch
ch = 'A'; //将变量 ch 的值赋为 'A'
```

当然我们也可以在声明的同时给变量赋值，因此上面的代码等价于：

```
int a = 5;
char ch = 'A';
```

4.更多的赋值运算符

除了使用`=`外，还有其它一些可以给变量赋值的运算符。

```
***
</> 示例 2-1 赋值运算符

1  #include <iostream>
2
3  using namespace std;
4
5  int a, b = 2; //声明了 2 个 int 型变量，并将 b 的值赋为 2
6  double c = 3.14; //声明 double 型变量 c，并将它赋为 3
7
8  int main()
9  {
10     a = 1; //将 a 的值赋为 1
11     a += 3; //将 a 的值增加 3
12     c /= 2; //将 c 除以 2 赋值
13     b--; //将 b 的值减去 1
14     cout << a << " " << b << " " << c; //输出 a,b,c（还没学）
15     return 0;
16 }
```

！注意 在还没有学习变量作用范围的情况下，变量在主函数外部或主函数内的开头声明。暂时不要声明到其它位置。

输出示例：

```
4 1 1.57
```

Ln14：可以把 a,b,c 的值依次输出出来，这将在下一弹学到。

Ln11-12：这两行等价于

```
a=a+3;
c=c/2;
```

写成 scratch 就如图 2。我们想让变量自己加、减、乘、除一个数的时候可以分别使用运算符`+=`，`-=`，`*=`，`/=`来进行。

Ln13：`++`是**自增运算符**，使变量的值增加 1。与之相反的是**自减运算符**`--`，使变量值减 1。格式为：

```
变量++;
变量--;
```

你也可以尝试在主函数外部声明一个 int 型变量，不给它赋值，并把它输出，会发现其初值为 0。



图 2

知识点 4 常量

在一段程序运行过程中，始终不发生改变的量，称之为**常量**。比如示例 2-1Ln5 的数字 2 就是一个常量，上一弹中的字符串"Hello,world!" 是常量。

C++中我们用修饰符 const 来修饰一个变量以限定其为只读。这样的变量，命名时通常会全部大写。格式为：

```
const 数据类型 变量名 = 常量表达式；
```

如：

```
const int N = 10；
```

这个知识将在后面学习一维数组时发挥用处，可以先记忆下来。

©孤言版权所有，未经许可，不得以任何形式和方式使用本文的任何内容（包括但不限于文字、图片等）。

第二版日期：2019.9.8

第三版日期：2020.5.9

第四版日期：2020.8.16

字数：2869

