

# 小马加编信息学教案(三)

## 数据类型与变量

- 一. 课程内容
- 二. 知识讲解
  - 1. C++ 中的数据类型
    - \* 1.1 什么是数据类型
    - \* 1.2 C++ 语言中的主要数据类型
    - \* 1.3 C++ 语言中主要数据类型的取值范围
  - 2. C++ 中的变量与常量
    - \* 2.1 什么是变量和常量
    - \* 2.2 变量的声明
    - \* 2.3 常量的声明
    - \* 2.4 变量名的命名规则
    - \* 2.5 变量的初始化与赋值
- 三. 经典例题
- 四. 提高巩固

### 一. 课程内容

1. C++ 中的数据类型
2. C++ 中的常量与变量
3. C++ 中的字符与字符串
4. C++ 强制类型转换

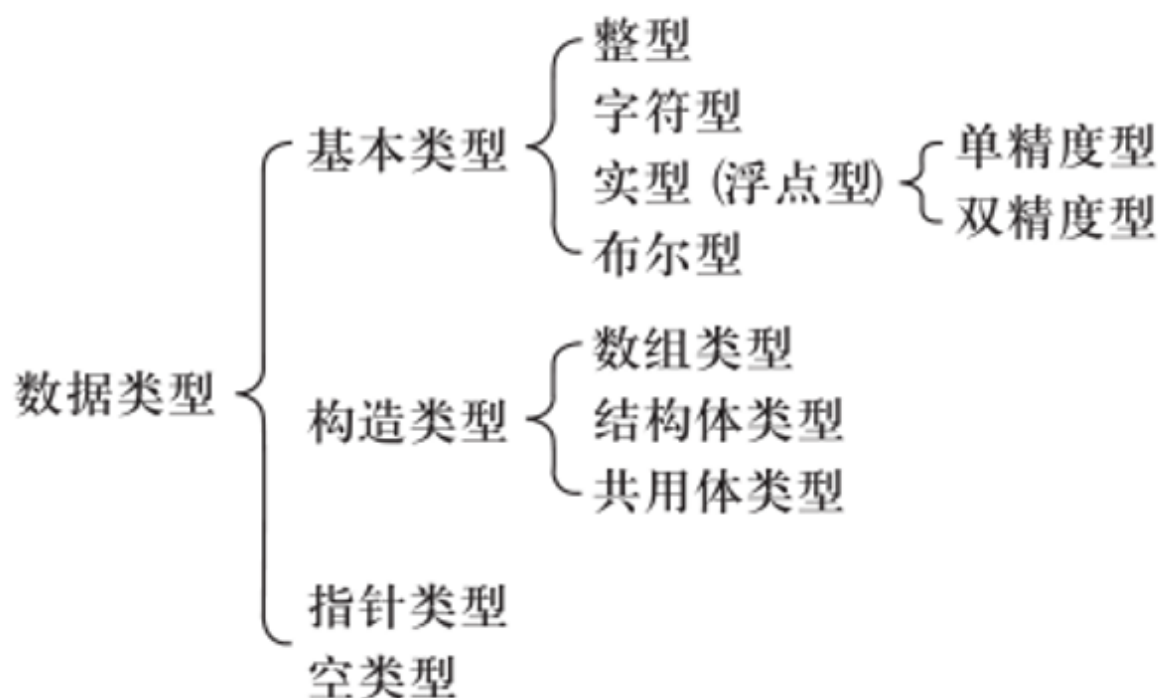
### 二. 知识讲解

#### 1. C++ 中的数据类型

##### 1.1 什么是数据类型

数据有数据类型、取值范围、运算类型三个要素。

## 1.2 c++ 语言中的主要数据类型



## 1.3 c++ 语言中主要数据类型的取值范围

表 2.2-1 数据类型及其字节长度、取值范围

类型	具体类型	字节长度	取值范围
整型	int	4(32 位)	-2147483648~2147483647
	short	2(16 位)	-32768~32767
	long	4(32 位)	-2147483648~2147483647
	long long	8(64 位)	$-2^{63} \sim 2^{63} - 1$
布尔型	bool	1(8 位)	0 或 1(严格意义上是非 0)
字符型	char	1(8 位)	-128~127
实型 (浮点型)	float	4(32 位)	-3.4E+38~3.4E+38, 6~7 位有效数字
	double	8(64 位)	-1.79E+308~1.79E+308, 15~16 位有效数字
	long double	不确定(与编译器有关)	-1.2E+4932~1.2E+4932, 18~19 位有效数字

由此可见，C++ 各种数据类型都有可能发现数据溢出的可能(即超出取值范围)，C++ 程序中出现这类错误时，编译和运行过程中并不报错。所以，需要特别细心，注意范围检查。

## 2. C++ 中的变量与常量

在程序中参与运算的量分为常量和变量

### 2.1 什么是变量和常量

**变量**是指在程序运行过程中，其值可以改变的量。

**常量**是在程序运行过程中，其值保持不变的量。

### 2.2 变量的声明

C++ 中定义一个变量的格式如下：

类型名 变量名1, 变量名2, ..., 变量名n;

ex:

```
int sum, total;
```

```
double area;
```

### 2.3 常量的声明

C++ 中定义一个符号常量的格式如下：

类型名 const 常量名;

或

const 类型名 常量名;

ex:

```
int const A = 100;
```

```
const float PI = 3.1415926;
```

使用符号常量是为了方便程序的修改，增强程序的读性。为了更清晰地区别常量和变量，**常量名通常用大写字母表示**。

在 C++ 程序中，使用的各种变量与常量都应“先声明，后使用”。

**变量与常量的声明**又会涉及数据类型、初始值和作用域等。

### 2.4 变量名的命名规则

程序中的常量名、变量名、函数名等都称之为**标识符**。

C++ 中的标识符命名规则如下：

- (1) 标识符不能和系统里的关键字同名。
- (2) 标识符只能包括字母、数字和下划线“\_”，并且开头只能是字母或下划线。
- (3) 标识符必须先定义、后使用。
- (4) 在同一作用域内，标识符不能重复定义。

(5) 标识符是区分大小写的，如 A 和 a、sum 和 Sum 都是不同的标识符。

(6) 标识符要尽量做到“见名知义”，增强程序的可读性。

练习：找出下面符合 C++ 规则的标识符

Mr.Wang, sum, Day, school\_name, a>=b, ¥1234

lotus\_1\_2, sum123\_, \_day, 234NUM, #34, int

符合 C++ 规则的标识符有：

sum, Day, school\_name, lotus\_1\_2, sum123\_, \_day。

## 2.5 变量的初始化与赋值

**变量在定义的同时，可以给它指定一个初始值，称为变量的初始化。**

在 C++ 中，可以通过“赋值”语句来修改变量的值。赋值语句的格式如下：

变量名 = 值或者表达式 ；

其中，“=”称为“赋值号”。

ex:

a = 5;

## 2.6 全局变量与局部变量

- 放在函数体外定义的变量叫做全局变量，它在声明定义之后可以在任何函数内使用。
- 如果全局变量未赋初值，则默认为0。
- 全局变量作用效果一直到程序结束为止。
- 放在函数体内定义的变量叫做局部变量，它在声明定义之后只可以在当前函数内使用。
- 如果局部变量未赋初值，则默认为随机值
- 局部变量作用效果随当前函数的结束而结束。

# 3. 字符与字符串

## 3.1 什么是字符、字符串

**存放单个字符的变量，称为字符变量。**

字符变量的值是用一对单引号括起来的一个字符，如 'a' 和 '?' 等。

存放一串有若干字符的变量，称为字符串变量。字符串变量的值是用一对双引号括起来的字符序列，如 "Good Morning!" 和 "a" 等。

如果双引号内什么都没有，称为空字符串，与双引号内有一个空格的字符串是不一样的。

## 3.2 字符变量

定义字符变量的格式如下：

char 标识符1, 标识符2, ..., 标识符n;

**字符变量在内存中占用 1 个字节。**

在 c++ 中，字符数据在内存中实际存储的是其 *ASCII* 码值。  
所以，从本质上说，**字符数据与整数数据的存储方式完全相同**

**可以使用 (int)a 将一个 char 类型的字符变量转换为 int 类型的整数**  
**也可以使用 (char)b 将一个 int 类型的整数变量转换为 char 类型的字符变量**

### 3.3 转义字符

c++ 中定义了一些字符前加 “\” 的特殊字符，称为转义字符

字符形式	含义	ASCII 码值(十进制)	字符形式	含义	ASCII 码值(十进制)
\n	换行	10	\"	双撇号字符"	34
\\	反斜杠字符\	92	\0	空操作字符	0
\'	单撇号字符'	39			

### 3.4 字符变量专属输入输出方法

#### getchar() 函数

```
getchar ();
```

用于读入一个字符变量

ex:

```
char ch;
ch = getchar();
```

#### putchar() 函数

```
putchar (输出字符);
```

用于输出一个字符变量

ex:

```
char ch = 'a';
putchar(a);
```

## 4. 强制类型转换

在 c++ 中可以进行“强制类型转换”,将某一个数据类型的数据强制转换为另一个数据类型的数据格式为:

( 类型名 )( 表达式 )

( 类型名 ) 变量

ex:

```
(double)a           //将a强制转换成double型
(int)(x + y)         //将x+y的值强制转换为int型
(float)(4 % 3)       //将4%3的值强制转换成float型
```

## 三. 经典例题

1. 阅读并上机调试运行以下程序，并预测、观察运行结果

```
#include<bits/stdc++.h>
using namespace std;

int main(){
    int a,b,c;
    double d,e;
    a = 5;b = 6;
    d = 2.53;
    c = a + b;
    e = c + d;
    cout << a << " + " << b << " = " << c << endl;
    cout << c << " + " << d << " = " << e << endl;
    return 0;
}
```

```
#include<bits/stdc++.h>
using namespace std;

int main(){
    int a = 2147483647;
    a = a + 1;
    cout << "a=" << a << endl;
    return 0;
}
```

```
#include<bits/stdc++.h>
using namespace std;

int main(){
    int b = 0, a = 100;
    char ch = 'A';
    b = a + ch;
    cout << ch << " " << int(ch) << " " << b << endl;
    return 0;
}
```

2. 输入一个英文字母，输出它的 $ASCII$ 码值

样例输入	样例输出
a	97
Z	90

3. 编写一个程序，定义两个浮点数变量  $a, b$ ，并读取依次输入的浮点数，输出  $a/b$  的值

样例输入	样例输出
2.2 1.1	2
11.25 4.5	2.5

## 四. 提高巩固

1. 输入一个数字，输出该数字 $ASCII$ 码值所对应的字母，输入保证有效。

样例输入	样例输出
97	a
90	Z

2. 输入一个英文字母，输出这个英文字母在字母表中后面的那个字母

样例输入	样例输出
a	b
G	H

3. 依次输入一个等腰梯形的上底、下底、高，输出它的面积(有可能为小数)

样例输入	样例输出
2 4 5	15
5 6 3	11.5