数据类型

//在创建一个变量或常量的时候，必须指出它的数据类型，以便于为其分配**合适的内存空间**。

前一节课我们简单地提过变量创建的语法：

//数据类型 变量名 (=初始值)

数据类型分类如下：

**1.整型**

整型又分为以下四种：

|  |  |  |
| --- | --- | --- |
| 类型 | 占用空间 | 取值范围【讲完（1）再写】 |
| short(短整型) | 2字节 | -2^15~2^15-1 （32768） |
| int（整型） | 4字节 | -2^31~2^31-1 |
| long（长整形） | windows为4字节 | -2^31~2^31-1 |
| long long（长长整型） | 8字节 | -2^63~2^63-1 |

//（1）**计算机内存的基本单元是位(bit)**，每个位可以看作一个开关，或者说可以发出两种不同信号的灯，通常下**1个字节是8位**，那我们想，8盏双新号彩灯，将它们进行排列组合，会有多少种不同的信号组合？显然是28种——也就是说**一个字节的内存最多可以表示28个不同的数**，那我们看比如int类型的变量，就可以表示2^32个不同的数，正负数平均一分，取值范围就是这么来的。

//（2）**“合适的内存空间”：用多少占多少**

（3）不要超限；(short a=32888;)

示例代码：

short a=32888;

cout<<"a="<<a<<endl;

int num1=2.5;

cout<<"num1="<<num1<<endl;

long num2=100.0;

cout<<"num2="<<num2<<endl;

**2.浮点型（实数）**

浮点型变量分为两种：

|  |  |  |
| --- | --- | --- |
| 类型 | 占用空间 | **包含整数部分**的有效数字位数 |
| float（单精度） | 4字节 | 7位有效数字 |
| double（双精度） | 8字节 | 15-16位有效数字 |

（13.14，四位有效数字）

实例代码1：

float f1=3.1415926；

cout<<”f1=”<<f1<<endl;

double d1=3.1415926；

cout<<”d1=”<<d1<<endl;

输出结果：

f1=3.14159

d1=3.14159

结论：默认情况下，浮点数最多可输出**6位**有效数字，更加精确的需要额外配置，这里暂时不作详细说明。

示例代码2：【科学计数法】

float f2=3e2 //3\*10^2

cout<<...

float f3=3e-1 //3\*10^(-1)

cout<<...

输出结果：

f2=300

f3=0.3

示例代码3：【可恶的除法】

int num1=7/4； （结果：num1=1，改成float和double后还是1）

float num2=7/4.0； //num2=1.75

float num3=7.0/4 //num3=1.75

**3.字符型**

字符型变量用于显示**单个**字符。

char ch=‘m’; //单引号！！！

字符型变量占用1个字节；存储单元中存储的是ASCII码；

示例代码：

char ch=’a’; (双引号、字符串错误演示)

cout<<ch<<endl;

cout<<ch+1<<endl;

cout<<(int)ch<<endl;

输出结果：

a; (A)

98; 66

97; 65

**4.转义字符**

以反斜杠\起始的特殊字符；

例如：\n换行； \t水平制表； [\\代表一个反斜线字符\;](\\\\代表一个反斜线字符\\;) \r回车；

示例代码：

cout<<”Hello C++\n”;

cout<<”录制不易，大家三连支持下呗！”<<endl; //c语言中没有endl

cout<<”\”<<endl; //一个反斜杠报错，必须两个才能输出

cout<<”lalala\tHello C++”<<endl; //\t与前面占八个字符，不够用空格补齐，适合做课程表，输出整齐

cout<<”lala\tHello C++”<<endl;

cout<<”aaa\tHello C++”<<endl; //再用空格试一试发现对不齐

**5.字符串**

用于表示一串字符（回忆char只能一个字符）

（1）C风格字符串(数组/指针)

char str1[]=“Hello C++”； //双引号

cout<<str1<<endl;

(2)C++风格字符串

string str2=“Hello C++”；

cout<<str2<<endl; //要包含头文件#include<string>

6.布尔类型bool

true——真，1；

false——假，0；

bool flag=true;

cout<<flag<<endl;

flag=false;

cout<<flag<<endl;

最后要三连