GESP 专题课（一级）第 1 课 基本数据类型与基本运算

一、基本数据类型

（一）基本数据类型概览

类型	关键字	大小（字节）	示例	用途
整型	int	4	-1, 0, 5	整数默认
浮点型	float	4	1.0f, -3.14f	单精度浮点数，需加 f 后缀
	double	8	1.0, -3.14	双精度浮点数，小数默认
字符型	char	1	'a', '0', '?'	单个字符，单引号括起
字符串型	string	长度可变	"apple"	字符串
布尔型	bool	1	true, false	逻辑值

（二）自动（隐式）类型转换

1. **基本概念**：编译器自动进行的类型转换，无需程序员显式指定。发生时机为不同类型数据混合运算、赋值、函数传参时。转换规则是低精度类型向高精度类型转换，顺序为 char→bool→int→long long→double。

2. **算术运算中的自动转换**

```C++
int a = 10;
double b = 3.14;
cout << a + b; //int 自动转为 double，a 转换为 double(10.0)，结果为 13.14
```

3. **赋值时的自动转换**

```C++
int x = 5;
double y = x; //int 自动转为 double，y=5.0
double pi = 3.14159;
int approx = pi; //double 自动转为 int，approx=3（截断小数）
```

（四）强制类型转换（显式类型转换）

```
C++
// C 风格强制转换：截断小数部分
double pi = 3.14159;
int intPi = (int)pi; // intPi 的值为 3
// C++风格强制转换：字符转 ASCII 码
char c = 'A';
int code = int(c); // 'A'的 ASCII 码为 65，code 的值为 65
```

二、基本运算

（一）算术运算符

运算符	作用	使用类型	考点
+	加法	整型、浮点型	-
-	减法	整型、浮点型	-
*	乘法	整型、浮点型	-
/	除法	整型、浮点型	整型除法：结果向下取整，舍弃小数部分；浮点型除法：保留小数，结果为浮点值
%	取余	仅整型	n%2==0 用于判断奇偶；a%b==0 用于倍数判断

运算顺序：

1）先算乘、除、取余，后算加、减；

2）同级运算符按从左到右顺序计算；

3）可通过()强制改变优先级，()内运算优先执行。

（三）逻辑&关系运算符

运算符	作用	返回结果	关键考点
>	判断左值大于右值	true / false	禁止连续使用>表示范围
<	判断左值小于右值	true / false	禁止连续使用<表示范围，数学中“3 < x < 5”，C++中需用 3 < x && x < 5
>=	判断左值大于等于右值	true / false	“大于”或“等于”满足其一即成立
<=	判断左值小于等于右值	true / false	“小于”或“等于”满足其一即成立
==	判断两值是否相等	true / false	区分赋值运算符=和等于运算符==
!=	判断两值是否不等	true / false	-
&&	全真则真，一假则假	true / false	短路求值：左边为假时，右边不再计算
||	一真则真，全假则假	true / false	短路求值：左边为真时，右边不再计算
!	真变假，假变真	true / false	单目运算符，优先级最高

（五）运算符的优先级

优先级	运算符
1	!
2	*、/、%
3	+、-
4	>、>=、<、<=
5	==、!=
6	&&
7	||

（七）顺序&赋值运算符

运算符	名称	作用	示例	等价表达式
=	基础赋值	将右值赋给左值（左值只能是 1 个变量）	a = 5	-
+=	加法赋值	左值 += 右值	a += 3	a = a + 3
-=	减法赋值	左值 -= 右值	a -= 2	a = a - 2
*=	乘法赋值	左值 *= 右值	a *= 4	a = a * 4
/=	除法赋值	左值 /= 右值	a /= 2	a = a / 2
%=	取余赋值	左值 %= 右值	a %= 3	a = a % 3
x++	后置自增	先使用 x 的原	int x = 5; x++	-

		值，再将 x 加 1		
++x	前置自增	先将 x 加 1，再使用 x 的新值	int x = 5; ++x	-
,	逗号运算符	连接多个表达式，按顺序执行，返回最后一个表达式的结果	int res = (a=2, b=3, a+b); cout << res; （输出 5）	-