

小马加编信息学教案(四)

赋值语句，运算符与运算规则

- 一. 课程内容
- 二. 知识讲解
 - 1. C++ 赋值语句
 - * 1.1 C++ 赋值语句的概念
 - * 1.2 C++ 赋值语句的应用
 - 2. C++ 算术运算符
 - 3. 表达式及其运算规则
 - 4. 复合运算符
 - * 4.1 算术复合
 - * 4.2 赋值复合
 - * 4.3 自加运算符和自减运算符
 - 5. 运算规则
 - * 5.1 自动类型转换
 - * 5.2 运算符的优先级与结合性
- 三. 经典例题
- 四. 提高巩固

一. 课程内容

1. C++ 赋值语句
2. C++ 基本算术运算符
3. C++ 复合运算符
4. C++ 运算规则

二. 知识讲解

1. C++ 赋值语句

1.1 C++ 赋值语句的概念

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 = 3;
```

需要注意的是：a=b 与 a == b 是完全不同的意义

“==” 是关系运算符，判断是否相等

赋值语句具有返回值，返回值即为该赋值语句成功赋值的内容

1.2 C++ 赋值语句的应用

尝试使用赋值语句实现以下功能：输入两个变量 a,b ,将他们的值交换后输出。

我们会发现，如果直接使用 a = b; 会将 a 原先的值被覆盖而丢失
所以我们**需要使用经典的"三变量交换法"**，类似于"倒水交换法"

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

int main() {
    int a, b;
    cin >> a >> b;
    int t; //t作为暂存变量
    t = a; //将a的值临时存放到t中
    a = b; //将b的值赋值给a
    b = a; //将t中暂存的原先的b值赋值给a

    //通过三条赋值语句的"三变量交换法"，就成功地将a, b的值进行了交换

    cout << a << ' ' << b << endl;
    return 0;
}
```

2. C++ 算术运算符

C++ 基本算术运算符如下

运算符	含义	说明	举例
+	加法	加法运算	$1+2=3, 1.5+2=3.5$
-	减法	减法运算	$5-4=1, 5.12-8=-2.88$
*	乘法	乘法运算	$4*5=20, 0.125*8=1.0$
/	除法	除法运算	$6/5=1, 5/2.0=2.5$ $-5.0/2=-2.5, -5.0/-2.0=2.5$
%	模	取余运算	$7\% 5=2, 7\%-5=2$ $-7\% 5=-2, -7\%-5=-2$
()	括号	改变运算的顺序	$2*(5+1)=12, 100/(8*2)=6$

3. 表达式及其运算规则

将常量、变量、算术运算符、括号以及函数连接在一起的计算式，称为算术表达式

当一个算术表达式中包含多个运算符时，遵循的 **计算顺序（运算优先级）** 为：先算“括号”里的，再算“乘法、除法、模”，最后算“加法、减法”，同一级别的按照类型从左往右依次计算。

表达式的计算结果称为表达式的值。

4. 复合运算符

在 c++ 中，赋值语句非常灵活，可形成多种**复合写法**。

4.1 算术复合

$a=a+x$; 写成 $a+=x$;
同理还有 $-=$ 、 $*=$ 、 $/=$ 、 $\% =$

4.2 赋值复合

c++ 支持在一个表达式里出现连续的赋值号，即一次给几个变量赋同一个值。
此时计算顺序是从右到左，有括号存在的，先计算括号内的表达式。

ex
 $a = b = 5$;
先将5赋值给 b ，($b = 5$) 返回的5值继续赋值给 a ,最后 a 与 b 都被赋值为 5

4.3 自加运算符和自减运算符

自加运算符 “++” 有两种写法，即 `i++` 或者 `++i`。

在单独使用时，其作用相同，都相当于`i=i+1`。

自减运算符 “--” 也有两种写法，即 `i--` 或者 `--i`。

需要特别注意的是，`++i` 是先进行 `i` 的自加操作，再返回 `i` 自加后的值

而 `i++` 是先返回 `i` 的值以后再进行自加操作

`--i` 与 `i--` 同理

5. 运算规则

5.1 自动类型转换

在 `c++` 中，整型数据和实型数据可以混合运算，字符型数据可以和整型数据通用。

但在具体计算时，编译系统会自动将不同类型的数据转换成同一类型，再进行运算

例如,在赋值运算中，赋值号两边数据的类型不同时，赋值号右边量或者表达式值的类型将自动转换为左边量的类型。

这一行为被称为数据类型的“自动转换”。



由此可见，整数除以整数，永远只能得到整数！

5.2 运算符的优先级与结合性

`c++` 语言还规定了运算符的优先级和结合性。

在表达式求值时，先按运算符的优先级别高低次序执行，例如先乘除后加减。

如果一个运算对象两侧的运算符的优先级别相同，如 `a - b + c` ,则按规定的“结合方向”处理。

算术运算符的结合方向为“先左后右”，又称之为“左结合性”。

有些运算符的结合方向为“自右向左”，即右结合性，例如赋值运算符。

`c++` 的运算符优先级与结合性表

优先级	运算符	结合性	优先级	运算符	结合性
1	()、[]	从左往右	5	<<、>>、>>>	从左往右
2	!、+(正)、-(负)、~、++、--	从右往左	6	<、<=、>、>=、instanceof	从左往右
3	*/、/、%	从左往右	7	==、!=	从左往右
4	+(加)、-(减)	从左往右	8	&(按位与)	从左往右
9	^	从左往右	12		从左往右
10		从左往右	13	?:	从右往左
11	&&	从左往右	14	=、+=、-=、*=、/=、%= &=、 =、^=、~=、<<=、 >>=、>>>=	从右往左

当然，我们只需要对上表有一个初步了解，能够正确掌握基本运算规则，并且能够计算表达式的值即可

实际上，自己进行编程时，涉及到多个优先级时应该尽可能加上括号，既能防止出错也能提高代码的可读性

三. 经典例题

1. 阅读下列程序，观察写出并上机检测程序运行结果

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

int main() {
    int i = 3, j;
    j = ++i;
    cout << j << i << endl;
    j = i++;
    cout << j << i << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;

int main(){
    int x;
    float a = 15.75;
    double b = 323.01;
    cout << (int) a << endl;
    cout << (int) (b + 0.5) % 10 << endl;
    cout << (int) b % (int) a + 1 << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;

int a, b;

int main() {
    cin >> a >> b;
    a += b;
    b -= 1;
    cout << a << " " << b << endl;
    a /= 2;
    b *= b;
    cout << a << " " << b << endl;
    a %= b;
    b %= 10;
    cout << a << " " << b << endl;
    return 0;
}
```

```
#include <iostream>
using namespace std;

int a, b;

int main(){
    a = b = 5;
    cout << a << ' ' << b << endl;
    (a = b) = 10;
    cout << a << ' ' << b << endl;
    (a += b) = 20;
    cout << a << ' ' << b << endl;
    return 0;
}
```

2. 存钱

小明将自己压岁钱存在了妈妈那，妈妈承诺在一个月后把钱还给小明并奖励小明 30% 的利息。小明想告诉你他存了多少钱，并希望你告诉他一个月后 he 可以从妈妈那拿到多少钱？

输入格式：一个正整数

输出格式：小明最后可以拿到的钱，可能含有小数

样例输入	样例输出
100	130
31	40.3

3. XR卖水果

苹果的价格是10元一斤

梨的价格是8元一斤

因为顾客同时买了两种水果所以XR将给这位顾客打八折！

输入格式：两个正整数， X ， Y 表示由一位顾客买了 X 斤苹果 Y 斤梨

输出格式：输出为一个整数，这位顾客这次买水果要付的钱
(数据保证结果一定是一个整数)

样例输入	样例输出
1 5	40

四. 提高巩固

1. 小玉买笔

班主任给小玉一个任务，到文具店里买尽量多的签字笔。

已知一只签字笔的价格是1元9角，而班主任给小玉的钱是 a 元 b 角，小玉想知道，她最多能买多少只签字笔呢？

输入格式: 一行内，包括两个整数，依次表示 a 和 b ， $a \leq 10000, b \leq 9$ 。

输出格式: 输出一个整数，表示小玉最多能买多少只签字笔。

样例输入	样例输出
10 3	5

2. 小鱼游泳

这一天，小鱼给自己的游泳时间做了精确的计时（本题中的计时都按 24 小时制计算）
它发现自己从 a 时 b 分一直游泳到当天的 c 时 d 分
请你帮小鱼计算一下，它这天一共游了多少时间呢？
小鱼游的好辛苦呀，你可不要算错了哦。

输入格式：一行内输入 4 个整数，分别表示 a, b, c, d

输出格式：一行内输出 2 个整数 e 和 f ，用空格间隔，依次表示小鱼这天一共游了多少小时多少分钟。其中表示分钟的整数 f 应该小于 60。

样例输入	样例输出
12 50 19 10	6 20

说明:

对于全部测试数据，

$$0 \leq a, c \leq 24, 0 \leq b, d \leq 60$$

且结束时间一定晚于开始时间。

3. 成绩(NOIP2017普及组第一题)

题目描述

牛牛最近学习了 c++ 入门课程，这门课程的总成绩计算方法是：总成绩 = 作业成绩 \times 20% + 小测成绩 \times 30% + 期末考试成绩 \times 50% 牛牛想知道，这门课程自己最终能得到多少分。

输入格式：

输入只有 1 行，包含三个非负整数 A 、 B 、 C ，分别表示牛牛的作业成绩、小测成绩和期末考试成绩。

相邻两个数之间用一个空格隔开，三项成绩满分都是 100 分。

输出格式：

输出格式：牛牛最后的成绩

样例输入	样例输出
100 100 80	90
60 90 80	79

说明

【输入输出样例 1 说明】

牛牛的作业成绩是 100 分，小测成绩是 100 分，期末考试成绩是 80 分，总成绩是 $100 \times 20\% + 100 \times 30\% + 80 \times 50\% = 20 + 30 + 40 = 90$

【输入输出样例 2 说明】

牛牛的作业成绩是 60 分，小测成绩是 90 分，期末考试成绩是 80 分，总成绩是 90 分，期末考试成绩是 80 分，总成绩是 $60 \times 20\% + 90 \times 30\% + 80 \times 50\% = 12 + 27 + 40 = 79$