

浮点数

身高5尺7寸？

- 美国人固执地使用英制计量单位，他们习惯用几尺几寸（英尺英寸）来报自己的身高。如果遇到一个美国人告诉你他是5英尺7，他的身高应该是一米几呢？
- $(5 + 7 \div 12) \times 0.3048 = 1.7018$ 米

计算身高的程序

```
Scanner in = new Scanner(System.in);  
System.out.print("请分别输入身高的英尺和英寸，如输入\"5 7\"表示5英尺7英寸：");  
int foot = in.nextInt();  
int inch = in.nextInt();  
System.out.println("身高是：" + ((foot +  
inch / 12) * 0.3048) + "米");
```

运行一下

- 为什么总是1.524米？

因为...

- 因为两个整数的运算的结果只能是整数
 - $10/3*3=>?$
- 10和10.0在Java中是完全不同的数
- 10.0是浮点数

插入

- 关于整数除法的结果

浮点数

- 带小数点的数值。浮点这个词的本意就是指小数点是浮动的，是计算机内部表达非整数（包含分数和无理数）的一种方式。另一种方式叫做定点数，不过在Java中你不会遇到定点数。人们借用浮点数这个词来表达所有的带小数点的数。

改进

- `(foot + inch / 12) * 0.3048;`



- `(foot + inch / 12.0) * 0.3048;`

- 当浮点数和整数放到一起运算时，Java会将整数转换成浮点数，然后进行浮点数的运算。

double

- inch是定义为int类型的变量，把int换成double，我们就把它改为double类型的浮点数变量了。
- double的意思是“双”，它本来是“双精度浮点数”的第一个单词，人们用来表示浮点数类型

浮点运算的精度

- 浮点计算是有误差的：
- `System.out.println(1.2-1.1);`



Android计算器低级错误？都是二进制惹的祸！

<http://www.guokr.com/article/27173/>

整数

- 整数类型不能表达有小数部分的数，整数和整数的运算结果还是整数。计算机里会有纯粹的整数这种奇怪的东西，是因为整数的运算比较快，而且占地方也小。其实人们日常生活中大量做的还是纯粹整数的计算，所以整数的用处还是很大的。

讨论

- 如果把程序里的inch变量换成double类型，那句语句后面的in.nextInt()还正确吗？

随堂

- 用一个变量meter来记录身高换算的结果，最后在System.out.println()里输出这个meter变量的值。

其他计算问题

运算符优先级

优先级	运算符	运算	结合关系	举例
1	+	单目取正	自右向左	$a * + b$
1	-	单目取负	自右向左	$a * - b$
2	*	乘法	自左向右	$a * b$
2	/	除法	自左向右	a / b
2	%	取余	自左向右	$a \% b$
3	+	加法	自左向右	$a + b$
3	-	减法	自左向右	$a - b$
3	+	字符串连接	自左向右	"hello" + "bye"
4	=	赋值	自右向左	$a = b$

如果记不住，就加括号 $()^{\wedge}_{-}$

单目运算符

- 只有一个算子的运算符：+、-

```
int a = 10;  
int b = -20;  
System.out.println(a * - b);
```


结合关系

- 一般自左向右
- 单目+-和赋值=自右向左

```
result = a = b = 3 + c;
```

```
result = 2;
```

```
result = (result = result * 2) * 6 * (result = 3 + result);
```

插入

- $6 + 5 / 4 - 2$
- $2 + 2 * (2 * 2 - 2) \% 2 / 3$
- $10 + 9 * ((8 + 7) \% 6) + 5 * 4 \% 3 * 2 + 3$
- $1 + 2 + (3 + 4) * ((5 * 6 \% 7 / 8) - 9) * 10$

强制类型转换

- 身高是1.524米
- `System.out.println("身高是: " + ((foot + inch / 12) * 0.3048 * 100) + "厘米");`
- 身高是152.4厘米

强制类型转换

- 如果想把一个浮点数的小数部分去掉，变成整数。

```
int i = 32 / 3.0;  
int i = (int)(32 / 3.0);
```

- (类型)值

身高?

- `System.out.println("身高是: " + (int)((foot + inch / 12) * 0.3048 * 100) + "厘米");`

强制类型转换

```
double b = 10.3;  
int a = (int)b;
```

- 只是从那个变量计算出了一个新的类型的值，它并不改变那个变量，无论是值还是类型都不改变

强制类型转换

```
double a = 1.0;  
double b = 2.0;  
int i = (int)a / b; int i = (int)(a / b);
```

- 强制类型转换的优先级高于四则运算

```
int a = 5;  
int b = 6;  
double d = (double)(a / b);
```