



常量的概念

下面的代码中，PI 和 3.14 都是常量。那究竟什么常量呢？

```
1. const double PI=3.14;
```

常量是在程序中使用的具体的数和字符。在程序运行过程中，其值不能被更改。

【整型常量】

整型常量是表示整数的常量。

	表示方法	示例
十进制形式	以非 0 数字开头，由 0~9 构成	如 10、-10
八进制形式	以数字 0 打头，由 0~7 构成	如 010，表示八进制整数 10
十六进制形式	以 0x 打头	如 0x1F，表示十六进制整数 1F

表1. 不同进制的整型常量表示法

【实型常量】

实型常量是表示实数的常量：

- 直接表示：如 3.14
- 科学记数法：-2.1E+2

【布尔常量】

布尔常量是表示逻辑结果的常量：

- true 表示“真”
- false 表示“假”



【字符常量】

字符常量是表示单个字符的常量。所有字符采用 ASCII 编码。在程序中，通常用一对单引号将单个字符括起来，如'b'、'A'、'5'、'%'等。

对于无法显示的控制字符和特殊字符，可以采用转义字符的方式进行表示。

转义字符	含义
'\n'	换行
'\t'	水平制表
'\b'	退格
'\r'	回车（不换行）
'\0'	空字符
'\"'	单引号
'\"'	双引号
'\\'	反斜杠字符
'\ddd'	1 ~ 3 位八进制数所代表的字符，例如'\101'就表示字符'A'
'\xhh'	1 ~ 2 位十六进制数所代表的字符，例如'\x41'就表示字符'A'

表2. 转义字符

【字符串常量】

字符串常量是表示字符序列的常量。是用双引号括起来的字符序列，如"abc"、"a"、""等。

特别需要注意的是，'a'与"a"表示的含义是不同的，'a'表示一个字符常量，"a"表示一个字符串常量。



符号常量的声明

常量可以直接使用，也可以给常量取个名字，用一个标识符来代表，这个名字就是符号常量。

符号常量声明的语法格式：

1. `const` 类型 常量名=常量字符串；

以下是合法的符号常量声明示例：

1. `const double PI=3.1415926;`
2. `const int HOUR_PER_DAY=24;`

在习惯上，常量名用大写，而变量名用小写，以便于区别。

和变量定义的位置不同，常量的定义一般放在 `int main()` 的外面。

```
1. const double PI = 3.1415926;
2. int main()
3. {
4.     ...
5.     return 0;
6. }
```

定义 `PI` 代表常量 3.1415926，计算机在编译程序时，遇到 `PI` 就用常量 3.1415926 代替，因此 `PI` 可以和常量一样进行运算。

使用符号常量有如下的好处：

- 增加了程序的可读性。见到 `PI` 就可知道它代表圆周率，定义符号常量名时应该尽量见名知意。
- 增加了程序的易改性。只需改动一处，程序中的所有 `PI` 都会自动全部代换，做到“一改全改”。



圆的计算

【问题描述】

输入半径 (radius) r ，求圆的周长 (perimeter) 及面积 (area)。

【输入样例】

1

【输出样例】

6.28319
3.14159

【参考程序】

```
1. #include<iostream>
2. using namespace std;
3. //PI 是符号常量，代表 3.1415926
4. const double PI = 3.1415926;
5. int main()
6. {
7.     //定义实型变量：r 半径
8.     double r;
9.     //显示提示符 radius=
10.    cout << "radius=";
11.    //输入 r 的值
12.    cin >> r;
13.    //计算圆的周长并显示计算结果
14.    cout << "perimeter=" << 2 * PI * r << endl;
15.    //计算圆的面积并显示计算结果
16.    cout << "area=" << PI * r * r << endl;
17.    return 0;
18. }
```

【说明】

可以根据计算精度的需求，定义 PI 的有效位数。

符号常量不能被再赋值。



圆柱体的表面积

【问题描述】

输入底面半径 r 和高 h ，输出圆柱体的表面积。

【输入样例】

3.5 9

【输出样例】

274.889

【分析】

圆柱体的表面积由 3 部分组成：上底面积、下底面积和侧面积。

根据平面几何知识，底面积 $= \pi r^2$ ，侧面积 $= 2\pi rh$ 。

由于上下底面积相等，所以：

$$\begin{aligned} & \text{表面积} \\ &= \text{底面积} \times 2 + \text{侧面积} \\ &= 2\pi r^2 + 2\pi rh \\ &= 2\pi r(r + h) \end{aligned}$$

【参考程序】

```
1. #include<iostream>
2. using namespace std;
3. const double PI = 3.1415926;
4. int main()
5. {
6.     //定义双精度实型
7.     double r, h;
8.     cin >> r >> h;
9.     //输出结果
10.    cout << 2 * PI * r * (r + h) << endl;
11.    return 0;
12. }
```



太阳光到达地球的时间

【问题描述】

日地距离 (Earth-Sun Distance) 其最大值为 15,210 万千米 (地球处于远日点); 最小值为 14,710 万千米 (地球处于近日点); 平均值为 14,960 万千米; 这就是一个天文单位。已知光在真空中的传播速度是 299792458 米/秒。

请计算远日点、近日点处太阳光到达地球的时间以及平均时间。

距离=速度×时间

【输出格式】

有 1 行, 三个浮点数。

【输出样例】

```
507.351 490.673 499.012
```

【分析】

光速需要多次参与运算, 可以设为一个常量。

利用公式: 距离=速度×时间的逆运算可以计算出 时间=距离/速度。

【参考程序】

```
#include<iostream>
using namespace std;
const int S = 299792458;
int main()
{
    double a = 15210.0 * 10000 * 1000; //远日点距离
    double b = 14710.0 * 10000 * 1000; //近日点距离
    double c = 14960.0 * 10000 * 1000; //天文单位
    double ta = a / S; //远日点光到达时间
    double tb = b / S; //近日点光到达时间
    double tc = c / S; //平均到达时间
    cout << ta << " " << tb << " " << tc;
    return 0;
}
```