



# 知识点多总结





- C语言基本的数学运算有加（+）、减（-）、乘（\*）、除（/）。
- 在C语言里，参与计算的数据的类型会影响运算结果。
- 如果运算符前后两个数全是整数，那么运算结果也是整数。即使结果里有小数，也会舍弃小数部分，只保留整数，例如， $3/4$ 会等于0。如果其中有一个是浮点数，那么运算结果会是浮点数，例如， $3/4.0$ 会等于0.75。
- %（取余运算符）用于计算两个整数相除后的余数。例如， $10 \% 3$ 会等于1。
- 注意，余数的符号跟被除数（即 % 左边的数）相同。例如， $-10 \% 3$ 的结果是-1， $10 \% -3$ 的结果是1。
- C语言的标准数学库 `math.h` 提供了许多数学相关的函数和常量。
- 使用 `math.h` 库时，需要在代码开头包含相应的头文件：`#include <math.h>`



- 其中，常见的数学函数有：

**sqrt**: 用于计算平方根。

$\text{sqrt}(x)$  表示计算 $x$ 的平方根，其中 $x$ 必须为非负数。

**pow**: 用于计算幂。

$\text{pow}(x, y)$  表示计算 $x$ 的 $y$ 次方， $x$ 和 $y$ 可以是任意实数。

**sin**、**cos**、**tan**: 用于计算角度的正弦、余弦和正切值。

$\text{sin}(x)$ 、 $\text{cos}(x)$ 、 $\text{tan}(x)$  分别表示计算 $x$ 的正弦、余弦和正切值，其中 $x$ 的单位是弧度。

角度 $y$ 转弧度： $y * \pi / 180$



**abs**: 用于计算整数的绝对值。

**abs(x)** 表示计算整数x的绝对值。

**fabs**: 用于计算浮点数的绝对值。

**fabs(x)** 表示计算浮点数x的绝对值。

**exp**: 用于计算以e为底的指数函数。

**exp(x)** 表示计算e的x次方。

**log**: 用于计算自然对数（以e为底）。

**log(x)** 表示计算x的以e为底的自然对数，其中x必须为正数。



题目1、有以下程序

```
#include <stdio.h>
#include <math.h>
main()
{
    int x = 1, y = 2;
    double z = pow(x, y) + sqrt(y);  pow(1, 2) + sqrt(2)
    printf("%.2f\n", z);           1      +  1.414  =  2.414
}
```

程序运行后的输出结果是 ( A ) 。

- A. 2.41
- B. 2.00
- C. 3.41
- D. 3.00

## 题目2、有以下程序

```
#include <stdio.h>
#include <math.h>

int main()
{
    int a = 1, b = 4, c = 2;
    double x = 10.5, y = 4.0, z;
    z = (a + b) / c + sqrt(y) * 1.2 / c + x;
    printf("%f\n", z);
}
```

程序运行后的输出结果是 ( A )

- A. 13.700000
- B. 14.000000
- C. 15.400000
- D. 14.900000

$$(1 + 4) / 2 + \sqrt{4.0} * 1.2 / 2 + 10.5$$





### 题目3、有以下程序

```
#include <stdio.h>
#include <math.h>

main() {
    int a = 5;
    double b = 4.5, c;
    c = pow(a, 2) + fabs(b - 2.0);    pow(5, 2) + fabs(4.5 - 2.0)
    printf("%f\n", c);                52      +      | 2.5 |      = 27.5
}
```

程序运行后的输出结果是 ( B )

- A. 29.500000
- B. 27.500000
- C. 32.500000
- D. 30.500000



$$\sqrt{|n^x + e^x|}$$

sqrt( fabs(pow(n,x)+pow(e,x)) )

题目4、若有代数式（其中e仅代表自然对数的底数，不是变量），  
则以下能够正确表示该代数式的C语言表达式是（ A ）

- A. sqrt(fabs(pow(n,x)+exp(x)))       $\text{exp}(x) = \text{pow}(e,x) = e^x$
- B. sqrt(fabs(pow(n,x)+pow(x,e)))
- C. sqrt(abs(n^x+e^x))
- D. sqrt(fabs(pow(x,n)+exp(x)))



题目5、下列程序段的输出结果是 3。

```
#include <stdio.h>
```

```
int main() {  
    int i = 19, j;  
    j = i % 4;          19 ÷ 4 = 4 .....3  
    printf("%d", j);  
}
```