

浮点型-数据类型



华清远见 | 创客学院 小美老师

浮点型

- 浮点型：用于存储小数，即数学中的实数

类型	关键字	长度	精确位数
单精度	float	4字节	6位
双精度	double	8字节	15-16位
长精度（C99新增）	long double	看编译器	精度不少于double的精度

浮点型

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

```
int main( )
```

```
{
```

```
float a;
```

```
double b;
```

```
long double c;
```

```
printf("a size = %lu\n", sizeof(a));
```

```
printf("b size = %lu\n", sizeof(b));
```

```
printf("c size = %lu\n", sizeof(c));
```

```
return 0;
```

```
}
```

```
linux@xiaomei:~/c_source$ gcc  
float_demo.c -Wall
```

```
linux@xiaomei:~/c_source$ ./a.out
```

```
a size = 4
```

```
b size = 8
```

```
c size = 16
```

浮点数的表示形式

浮点数

指数形式

0.008 \longrightarrow 8E-3

8000 \longrightarrow 8E+3

浮点数的表示形式

```
#include <stdio.h>

int main()
{
    float f1 = 0.008;
    float f2 = 8E-3;
    float f3 = 8000;
    float f4 = 8E+3;
    printf("f1=%f,f2=%f\n", f1, f2);
    printf("f3=%f,f4=%f\n", f3, f4);
    return 0;
}
```

```
$ gcc float_demo2.c -Wall
$ ./a.out
f1=0.008000,f2=0.008000
f3=8000.000000,f4=8000.000000
```

不论是浮点数形式，还是指数形式，输出效果是相同的

浮点型

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

```
int main( )
```

```
{
```

```
float a = 1.12345678901234567890;
```

```
double b =  
1.12345678901234567890;
```

```
long double c =  
1.12345678901234567890;
```

```
printf("a = [%.30f]\n", a);
```

```
printf("b = [%.30lf]\n", b);
```

```
printf("c = [%.30Lf]\n", c);
```

```
return 0;
```

```
}
```

绝大多数的浮点数是不能精确表示的

```
$ gcc float_demo3.c -Wall
```

```
$ ./a.out
```

```
a = [1.123456835746765136718750000000]
```

```
b = [1.123456789012345691247674039914]
```

```
c = [1.123456789012345691247674039914]
```

扫一扫，获取更多信息



THANK YOU