

# C 标准库 - <float.h>

## 简介

C 标准库的 `float.h` 头文件包含了一组与浮点值相关的依赖于平台的常量。这些常量是由 ANSI C 提出的，这让程序更具有可移植性。在讲解这些常量之前，最好先弄清楚浮点数是由下面四个元素组成的：

组件	组件描述
S	符号 ( +/- )
b	指数表示的基数，2 表示二进制，10 表示十进制，16 表示十六进制，等等...
e	指数，一个介于最小值 <code>e<sub>min</sub></code> 和最大值 <code>e<sub>max</sub></code> 之间的整数。
p	精度，基数 b 的有效位数

基于以上 4 个组成部分，一个浮点数的值如下：

$$\text{floating-point} = (S) p \times b^e$$

或

$$\text{floating-point} = (+/-) \text{precision} \times \text{base}^{\text{exponent}}$$

## 库宏

下面的值是特定实现的，且是通过 `#define` 指令来定义的，这些值都不得低于下边所给出的值。请注意，所有的实例 FLT 是指类型 `float`，DBL 是指类型 `double`，LDBL 是指类型 `long double`。

宏	描述
FLT_ROUNDS	定义浮点加法的舍入模式，它可以是下列任何一个值： <ul style="list-style-type: none"><li>-1 - 无法确定</li><li>0 - 趋向于零</li><li>1 - 去最近的值</li><li>2 - 趋向于正无穷</li><li>3 - 趋向于负无穷</li></ul>
FLT_RADIX	这个宏定义了指指数表示的基数。基数 2 表示二进制，基数 10 表示十进制，基数 16 表示十六进制。
FLT_MANT_DIG	这些宏定义了 FLT_RADIX 基数中的位数。

DBL_MANT_DIG LDBL_MANT_DIG	
FLT_DIG 6 DBL_DIG 10 LDBL_DIG 10	这些宏定义了舍入后不会改变表示的十进制数字的最大值（基数 10）。
FLT_MIN_EXP DBL_MIN_EXP LDBL_MIN_EXP	这些宏定义了基数为 FLT_RADIX 时的指数的最小负整数值。
FLT_MIN_10_EXP -37 DBL_MIN_10_EXP -37 LDBL_MIN_10_EXP -37	这些宏定义了基数为 10 时的指数的最小负整数值。
FLT_MAX_EXP DBL_MAX_EXP LDBL_MAX_EXP	这些宏定义了基数为 FLT_RADIX 时的指数的最大整数值。
FLT_MAX_10_EXP +37 DBL_MAX_10_EXP +37 LDBL_MAX_10_EXP +37	这些宏定义了基数为 10 时的指数的最大整数值。
FLT_MAX 1E+37 DBL_MAX 1E+37 LDBL_MAX 1E+37	这些宏定义最大的有限浮点值。
FLT_EPSILON 1E-5 DBL_EPSILON 1E-9 LDBL_EPSILON 1E-9	这些宏定义了可表示的最小有效数字。
FLT_MIN 1E-37 DBL_MIN 1E-37 LDBL_MIN 1E-37	这些宏定义了最小的浮点值。

## 实例

下面的实例演示了 float.h 文件中定义的一些常量的使用。

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

int main()
```

```
{  
    printf("The maximum value of float = %.10e\n", FLT_MAX);  
    printf("The minimum value of float = %.10e\n", FLT_MIN);  
  
    printf("The number of digits in the number = %.10e\n", FLT_MANT_DIG);  
}
```

让我们编译和运行上面的程序，这将产生下列结果：

```
The maximum value of float = 3.4028234664e+38  
The minimum value of float = 1.1754943508e-38  
The number of digits in the number = 7.2996655210e-312
```

[← C 标准库 – <errno.h>](#)[C 标准库 – <setjmp.h> →](#)[✍ 点我分享笔记](#)