

# 可变参数函数定义

——C 语言怪谈系列

```
//C99 Code
#include <assert.h>
#include <ctype.h>
#include <errno.h>
#include <float.h>
#include <limits.h>
#include <locale.h>
#include <math.h>
#include <setjmp.h>
#include <signal.h>
#include <stdarg.h>
#include <stddef.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <iso646.h>
#include <wchar.h>
#include <wctype.h>
#include <complex.h>
#include <fenv.h>
#include <inttypes.h>
#include <stdbool.h>
#include <stdint.h>
#include <tgmath.h>
```

```
//1, 函数声明或原型定义中, 第一个参数必须完整,
//最后一个参数必须是 3 个点代替省略的参数队列;
extern void simple_va_fun(int i, ...);
```

```
int main (int argc, char *argv[])
{
    simple_va_fun(100);
    simple_va_fun(100,200);

    return 0;
}
```

```
void simple_va_fun(int i, ...)
```

```

{
    va_list arg_ptr;//2, 定义 va_list 型变量 arg_ptr
    va_start(arg_ptr, i);//3, 用 va_start 宏初始化变量 arg_ptr,
        //宏的第二个参数是第一个可变参数的前一个参数,
        //是一个固定的参数。
    int j = va_arg(arg_ptr, int);//4, 用 va_arg 返回可变的参数,并赋值给整数 j,
        //va_arg 的第二个参数是你要返回的参数的类型。
    va_end(arg_ptr);//5, 用 va_end 宏结束可变参数的获取。

    printf("%d %d\n", i, j);//6, 在函数里使用可变参数序列 j。
    return;
}
//问题: 如果不使用可变参数宏, 怎么实现?
//或者可变参数宏是如何实现的?
//编程疯子 QQ52014481
//2011 年 10 月 23 日 河南信阳 整理

```