

头文件: `#include <stdio.h>`

`remove()` 函数用于删除指定的文件, 其原型如下:

```
int remove(char * filename);
```

【参数】`filename` 为要删除的文件名, 可以为一目录。如果参数 `filename` 为一文件, 则调用 `unlink()` 处理; 若参数 `filename` 为一目录, 则调用 `rmdir()` 来处理。

【返回值】成功则返回0, 失败则返回-1, 错误原因存于 `errno`。

错误代码:

1. EROFS 欲写入的文件为只读文件。
2. EFAULT 参数 `filename` 指针超出可存取内存空间。
3. ENAMETOOLONG 参数 `filename` 太长。
4. ENOMEM 核心内存不足。
5. ELOOP 参数 `filename` 有过多符号连接问题。
6. EIO I/O 存取错误。

【实例】下面的程序演示了如何使用 `remove()` 函数删除文件。

```
1. #include<stdio.h>
2. int main(){
3.     char filename[80];
4.     printf("The file to delete:");
5.     gets(filename);
6.     if( remove(filename) == 0 )
7.         printf("Removed %s.", filename);
8.     else
9.         perror("remove");
10. }
```

运行上述程序, 首先声明用于保存文件名的字符数组变量, 从控制台获取文件名, 然后删除该文件, 并根据删除结果输出相应的提示信息。

注意：实际开发中尽量避免使用gets()函数，gets()会影响程序的安全性和健壮性，请查看：[C语言gets\(\)函数：从流中读取字符串](#)