

# memcpy

## 函数原型

```
void *memcpy(void*dest, const void *src, size_t n);
```

## 功能

由 src 指向地址为起始地址的连续 n 个字节的数据复制到以 destin 指向地址为起始地址的空间内。

## 头文件

```
#include<string.h>
```

## 返回值

函数返回一个指向 dest 的指针。

## 说明

1. source 和 destin 所指内存区域不能重叠，函数返回指向 destin 的指针。
2. 与 strcpy 相比，memcpy 并不是遇到'\0'就结束，而是一定会拷贝完 n 个字节。

memcpy 用来做内存拷贝，你可以拿它拷贝任何数据类型的对象，可以指定拷贝的数据长度；

例：

```
char a[100], b[50];
```

```
memcpy(b, a, sizeof(b)); //注意如用 sizeof(a)，会造成 b  
的内存地址溢出。
```

strcpy 就只能拷贝字符串了，它遇到'\0'就结束拷贝；例：

```
char a[100], b[50];
```

```
strcpy(a,b);
```

3. 如果目标数组 destin 本身已有数据，执行 memcpy（）后，将覆盖原有数据（最多覆盖 n）。如果要追加数据，则每次执行 memcpy 后，要将目标数组地址增加到你要追加数据的地址。

//注意，source 和 destin 都不一定是数组，任意的可读写的空间均可。