当所指向的对象被释放或者收回，但是对该指针没有作任何的修改，以至于该指针仍旧指向已经回收的内存地址，此情况下该指针便称**悬垂指针**（也叫**迷途指针**）。

**悬垂指针**的成因：

在许多编程语言中（比如C），显示地从内存中删除一个对象或者返回时通过销毁栈帧，并不会改变相关的指针的值。该指针仍旧指向内存中相同的位置，即使引用已经被删除，现在可能已经挪作他用。

一个简单的例子：

{

char \*dp = NULL;

/\* ... \*/

{

char c;

dp = &c;

}

/\* c falls out of scope \*/

/\* dp is now a dangling pointer \*/

}