**我们不要创建局部变量的指针**

void processAdoptions(istream& dataSource)

{

while (dataSource) {

ALA \*pa = readALA(dataSource);

pa->processAdoption();

delete pa;

}

}

**注意如果**pa->processAdoption(); **发生错误，那么程序将无法执行下面的delete pa;所以造成内存泄漏**

**解决方法就是：具体方法是用一个对象代替指针 pa，这个对象的行为与指针相似，当对象离开作用域时候就会自动执行释放函数。我们可以使用c++的一个简单类模板** auto\_ptr，**解决此问题**

template<class T>

class auto\_ptr {

public:

auto\_ptr(T \*p = 0): ptr(p) {} **// 保存 ptr，指向对象**

~auto\_ptr() { delete ptr; } **// 删除 ptr 指向的对象**

private:

T \*ptr; // raw ptr to object

};

auto\_ptr<ALA> pa(readALA(dataSource)); **//这样我们就可以解决此问题**

**因为我们使用的是delete 指针 如果我们先要删除数组所以我们要自己重写**