

# 数组

## 题目描述

[刷新](#)

初学编程的小明觉得C++的数组太难用了。数组长度必须是一个数字或者const类型变量；如果想对数组整体进行复制，还得用循环处理。因此，你想到可以用面向对象的方法，将数组封装成对象，方便小明的操作。

你需要实现类 Array 满足以下条件：

- 构造时使用int变量指明数组长度，此后数组长度不会发生改变
- Array的元素为Node类的对象
- 能够通过operator[]访问数组元素，可以读取和修改
- 实现数组的复制构造，赋值操作
- 不出现内存泄露的问题

你需要编写 Array.h 和 Array.cpp 文件，打包压缩为一个文件，和我们已经写好的文件一同进行编译。

我们将使用测试函数来判断你的实现是否正确，核心的测试代码在 main.cpp 的 test 函数中。该函数从标准输入读入，处理并产生相应输出，最后由输出结果判定你的结果是否正确。

在 test 函数执行结束后，我们会调用 Node::outputResult() 函数。该函数会输出所有Node类型的对象被**普通构造**、**拷贝构造**、**移动构造**、**拷贝赋值**、**移动赋值**、**析构**的次数。

若你的程序没有内存泄露，则输出会满足：

普通构造 + 拷贝构造 + 移动构造 = 析构

我们知道，构造、拷贝构造、拷贝赋值所消耗的时间较长，移动构造、移动赋值所使用的时间较短。为了证明你的实现具有较好的效率，我们需要你的输出满足以下条件：

$(\text{普通构造} + \text{拷贝构造} + \text{拷贝赋值}) * 10 + (\text{移动构造} + \text{移动赋值}) \leq \text{参考答案}$

我们按以下方式对你的程序评分：

- 当你的程序通过测试，但存在内存泄露，获得30%的分数
- 当你的程序通过测试，不存在内存泄露，但不满足效率要求，获得70%的分数
- 当你的程序通过测试，不存在内存泄露，满足效率要求，获得100%的分数

注意：给出的参考答案并不是最优的结果，所以你并不一定要优化到最优。实际上，只要你正确实现了数组的移动构造，移动赋值操作，即可达到满分要求。

## 输入格式

第一行一个整数  $reference_{ans}$ ，代表参考答案。

接下来为测试函数的输入，具体含义请阅读测试代码。

第二行为  $len, xpos, xval, ypos, yval$

第三行为  $len$  个整数，读入到数组A

第四行为一个整数， $xval_{new}$

第五行为  $len$  个整数，读入到数组A

第六行为  $len$  个整数，读入到数组A

## 输出格式

前9行为测试函数输出，每行  $len$  个整数。

第10行为6个整数，分别代表Node类型的对象被**普通构造**、**拷贝构造**、**移动构造**、**拷贝赋值**、**移动赋值**、**析构**的次数。

# 样例输入

```
181
3 0 10 1 20
1 2 3
1000
4 5 6
7 8 9
```

# 样例输出

```
10 20 3
10 20 3
10 20 3
4 5 6
10 20 3
7 8 9
4 5 6
4 5 6
7 8 9
11 0 0 7 1 11
```

# 样例解释

该程序调用Node类**普通构造**、**拷贝构造**、**移动构造**、**拷贝赋值**、**移动赋值**、**析构**的次数为11、0、0、7、1、11次。  
普通构造 + 拷贝构造 + 移动构造 = 析构  
没有内存泄露  
(普通构造 + 拷贝构造 + 拷贝赋值) \* 10 + (移动构造 + 移动赋值) = 181  
刚好满足参考答案，故能获得满分。

# 数据范围与约定

$1 \leq len \leq 1000$   
所有出现的数在int范围内  
时限1s，内存限制256M。  
一共两个数据点，第一个数据点为样例，第二个数据点len=1000

# 提交文件

将 Array.h 和 Array.cpp 文件打包压缩为一个zip文件  
我们会将你提交的文件和我们预先设置好的文件一起编译运行。 文件下载：[下载链接](#)  
(/staticdata/problem/1967.N0UNPP02PtLsfX2.pub/kPSxTkGcNLIUVxNo.%E5%BD%92%E6%A1%A3.zip/%E5%BD%92%E6%A1%A3.zip)

# 特别提示

你可以使用STL，即OJ能编译通过的均能使用。

# 评分标准

- 当你的程序通过测试，但存在内存泄露，获得30%的分数
  - 当你的程序通过测试，不存在内存泄露，但不满足效率要求，获得70%的分数
  - 当你的程序通过测试，不存在内存泄露，满足效率要求，获得100%的分数
- 考试100%为OJ评分。

语言和编译选项

#	名称	编译器	额外参数	代码长度限制
0	custom	make		65536 B

递交历史

#	状态	时间	
329025	Accepted	2025-04-09 22:20:26	
329023	Compilation Error	2025-04-09 22:20:06	1
当前没有提交权限!			