

# 简单的Map实现

## 题目描述

刷新 

你需要在这份代码 (/staticdata/problem/2164.yRBkXzp9nyuQV7b6.pub/dFfAtmlmhr0lzR5o.download.zip/download.zip)的基础上帮忙完善 Map 类的实现（补充 Map.h 和 Map.cpp）。Map 类是一个能够实现 std::string 到 int 的映射的容器类，具有如下特性：

- 一个 Map 对象含有两个数据成员如下：

```
#pragma once
#include "Pair.h"

class Map{
    Pair * data;
    int sz;
public:
    Map(int n);
    // TODO
};
```

构造函数 Map(int n); 的形参 int n 指示被创建的 Map 对象需要能够容纳不多于 n 个键值对，一个键值对由 Pair 类的一个对象维护。你需要在构造函数内开辟一个大小为 n 个 Pair 对象的数组，并让数据成员 data 指向该数组的首位。

数据成员 sz 表明 Map 对象实际包含的键值对数量（必有  $0 \leq sz \leq n$ ）。

Map 类需要正确管理其动态分配的内存资源，确保在对象生命周期结束时释放所有分配的资源。

- 假定 s 是一个 string 对象，map 是一个 Map 对象，则调用 map[s] 满足如下性质：

	map 是常量	map 不是常量
s 是 map 的一个键	返回 s 对应的值，只能读，不能写	返回 s 对应的值，能读写
s 不是 map 的一个键	返回默认值 0	将 s 和默认值 0 配对添加到 map 中，并返回 s 对应的值，能读写

## 输入输出

- main.cpp 文件读取输入并构造一个 Map 类对象 map（及其常量引用 cmap）进行测试；
- 输入第一行是两个用空格隔开的整数：n 和 k，其中 n 表示被测试的 Map 对象的最大容纳量为 n 个键值对， $n > 0$ ；
- 输入后面是 k 行，每行是如下三种格式中的一种：
  - 1 [key]：查询并输出 map 中键为 [key]（字符串）对应的值，输出占一行
  - 2 [key] [val]：将 map 中键为 [key] 的值改为 [val]（整数）
  - 3 [key]：查询并输出 cmap 中键为 [key]（字符串）对应的值，输出占一行
- 最后输出 map 实际包含的键值对数量。

## 输入样例

```
30 8
1 abc
2 pwb 37068
2 ddddd 30771
3 abc
1 pwb
3 oop
2 abc 4448
2 ddddd 27157
```

## 输出样例

```
0
0
37068
0
3
```

## 提交格式

- 你需要提交多个文件，包含 Map.h 和 Map.cpp，可以不包括提供的 Map.h 之外的其他文件。

- 你应该将你的文件打包成一个zip压缩包并上传。**注意：你的文件应该在压缩包的根目录下，而不是压缩包的一个子文件夹下。**评测时，OJ会将除 Map.h 和 Map.cpp 之外的评测所需的文件（包括 main.cpp 等）贴入你的目录下进行编译并执行。

语言和编译选项

#	名称	编译器	额外参数	代码长度限制
0	oop_custom	make		1048576 B

递交历史

#	状态	时间
343684	Accepted	2025-05-18 16:50:04

1

递交答案

语言和编译选项

oop\_custom

▼

1

提交

文件请拖入编辑器中，或

上传文件