

## CCF全国信息学奥林匹克联赛（NOIP2016）复赛

## 提高组 day2

（模拟试题 - 北京大学 李煜东）

## 一. 题目概况

中文题目名称	图书列表	量化交易	Genius ACM
英文题目与文件名	booklist	trade	geniusacm
输入文件名	booklist.in	trade.in	geniusacm.in
输出文件名	booklist.out	trade.out	geniusacm.out
每个测试点时限	1 秒	1 秒	2 秒
内存限制	256 MB	256 MB	512 MB
测试点数目	10	10	10
每个测试点分值	10	10	10
结果比较方式	逐字节比较	全文比较（过滤行末 空格及文末回车）	全文比较（过滤行末 空格及文末回车）
题目类型	传统	传统	传统

## 二. 提交源程序文件名

对于 C++ 语言	booklist.cpp	trade.cpp	geniusacm.cpp
对于 C 语言	booklist.c	trade.c	geniusacm.c
对于 Pascal 语言	booklist.pas	trade.pas	geniusacm.pas

## 三. 编译命令

对于 C++ 语言	g++ booklist.cpp -o booklist -Wl,--stack=8388608	g++ trade.cpp -o trade -Wl,--stack=8388608	g++ geniusacm.cpp -o geniusacm -Wl,--stack=8388608
对于 C 语言	gcc booklist.c -o booklist -Wl,--stack=8388608	gcc trade.c -o trade -Wl,--stack=8388608	gcc geniusacm.c -o geniusacm -Wl,--stack=8388608
对于 Pascal 语言	fpc booklist.pas	fpc trade.pas	fpc geniusacm.pas

## 图书列表

### 【问题描述】

Peking University Library 的历史和 Peking University 一样长，它始建于 1898 年。截止到 2015 年，它包含大约 11000 千册馆藏图书，其中 8000 千册为纸质图书，其余为电子图书。毛泽东主席曾在 1918~1919 年间在该馆任职，他的工资是 8 大洋/月，当时顶尖教授的工资是 280 大洋/月。

现在小 G 在馆中担任与毛泽东曾经任职过的相同的职位。他的第一份工作是重新安排一些图书。他得到了一张列表，每个表项具有以下格式：

CATEGORY1/CATEGORY 2/.../CATEGORY n/BOOKNAME

这表示图书 BOOKNAME 位于目录 CATEGORY n 下，目录 CATEGORY n 位于目录 CATEGORY n-1 下，目录 CATEGORY n-1 位于目录 CATEGORY n-2 下，以此类推。也就是说，每个表项是由最后的一本图书，以及该图书所属的若干目录按照层级依次组成的。我们称 CATEGORY1 为一级目录，而 CATEGORY 2 为二级目录，以此类推。例如：

MATH/GRAPH THEORY

ART/HISTORY/JAPANESE HISTORY/JAPANESE ACIENT HISTORY

ART/HISTORY/CHINESE HISTORY/THREE KINDOM/RESEARCHES ON LIUBEI

ART/HISTORY/CHINESE HISTORY/CHINESE MORDEN HISTORY

ART/HISTORY/CHINESE HISTORY/THREE KINDOM/RESEARCHES ON CAO CAO

小 G 认为这份列表很不容易阅读和查找，于是他决定按照以下规则制作一份新列表，用缩进来体现图书与目录之间的层级关系：

- 1) n 级目录之前有  $4 \times (n-1)$  个空格的缩进。
- 2) 直接隶属于 n 级目录的图书前有  $4 \times n$  个空格的缩进。
- 3) 直接隶属于目录 X 目录与图书按照字典序列在目录 X 之后，但所有目录位于所有图书之前。
- 4) 所有一级目录按照字典序先后列出。

例如，上面的列表转化后将变为：

ART

    HISTORY

        CHINESE HISTORY

            THREE KINDOM

                RESEARCHES ON CAO CAO

RESEARCHES ON LIUBEI  
CHINESE MORDEN HISTORY  
JAPANESE HISTORY  
JAPANESE ACIENT HISTORY  
MATH  
GRAPH THEORY

请写一个程序帮助小 G 完成这项工作。

### 【输入格式】

输入原列表，共包含不超过 30 本图书，以一个数字 0 结尾。

每行列出一个表项，表项是一个由大写字母、数字、“/”和空格构成的字符串，长度不超过 100。

一本图书可能在列表中出现多次，但在转化后的列表中，它应该只出现一次。但是若同名的图书或目录若在不同的目录结构下，则认为他们是不相同的。换句话说，一个图书或目录由以它的名字为结尾的前缀唯一确定。

### 【输出格式】

输出新列表。本试题采用逐字节比较，行末请勿输出多余空格，文末以恰好一个换行符结尾。

样例输入 1	样例输出 1
B/A B/A B/B 0	B A B
样例输入 2	样例输出 2
A1/B1/B32/B7 A1/B/B2/B4/C5 A1/B1/B2/B6/C5 A1/B1/B2/B5 A1/B1/B2 A1/B1/B2/B1 A1/B3/B2 A3/B1 A0/A1 0	A0 A1 A1 B B2 B4 C5 B1 B2 B6 C5 B1 B5

	<div>B32</div> <div>B7</div> <div>B2</div> <div>B3</div> <div>B2</div> <div>A3</div> <div>B1</div>
--	--

【数据规模与约定】

- 对于 20%的数据，只有一级目录。
- 对于另外 20%的数据，没有同名的图书或目录。
- 对于另外 20%的数据，每本图书仅出现一次。
- 对于 100%的数据，参见输入格式中给出的数据范围，没有其它特殊约定。

# 量化交易

## 【问题描述】

applepi 训练了一个可以自动在股票市场进行量化交易的模型。通常来说，applepi 写出的模型，你懂得，就好比一架印钞机。不过为了谨慎起见，applepi 还是想先检查一下模型的效果。

applepi 收集了“塞帕思股份(surpass)”在最近的连续  $N$  天内的价格。在每一天中，他可以做如下事情之一：

1. 睡（把）觉（妹）。
2. 以当天的价格作为成交价买入 1 股“塞帕思”的股票。
3. 以当天的价格作为成交价卖出 1 股“塞帕思”的股票。

最初 applepi 不持有该股票。现在你需要计算出在最优策略下， $N$  天后 applepi 能够获得的最大利润。为了维护森林的和平，本着清仓甩锅的原则，在  $N$  天的交易结束后 applepi 也不能持有“塞帕思”的股票。

## 【输入格式】

每个测试点包含若干组数据，以 EOF 结尾。对于每组数据：

第一行 1 个整数  $N$ 。

第二行  $N$  个正整数，相邻两个整数之间用 1 个空格隔开，表示每一天股票的价格。

## 【输出格式】

对于每组数据，首先按样例所示的格式“Case #k:”输出该组数据的编号，然后输出一个整数，表示 applepi 最大能够获得的利润。

样例输入 1	样例输出 1
6 2 6 7 3 5 6 8 1 2 3 4 5 6 7 8	Case #1: 8 Case #2: 16
样例输入 2	样例输出 2
10 15831 47573 60015 51368 32460 34125 43074 75172 54014 93578	Case #1: 161084

**【数据规模与约定】**

对于 50%的数据， $1 \leq N \leq 1000$ 。

对于 100%的数据， $1 \leq N \leq 100000$ ，股票价格不超过 100000，每个测试点至多包含 5 组数据。

# Genius ACM

## 【问题描述】

**Advanced CPU Manufacturer** (ACM) is one of the best CPU manufacturers in the world. 每天, 该公司生产  $n$  台 CPU 并销售到世界各地。

ACM 公司的质检部门会对生产出的 CPU 进行成组测试, 对一组 (若干个) CPU 进行测试的方法如下:

- 1) 随机从该组 CPU 中选取  $m$  对 (即  $2m$  台), 若总数不足  $2m$  台, 则选取尽量多对。
- 2) 对于每一对 CPU, 测量它们之间的 **Relative Performance Difference** (RPD), 并把第  $i$  对的 RPD 记为  $D_i$ 。RPD 的计算方法在后面给出。
- 3) 该组 CPU 的 **Sqared Performance Difference** (SPD) 由以下公式给出:

$$SPD = \sum_i D_i^2$$

- 4) 该组 CPU 通过质检, 当且仅当  $SPD \leq k$ , 其中  $k$  是给定常数。

ACM 公司生产的 CPU 性能很好, 而质检部门制定的标准更是过于严格。通常他们把  $n$  台 CPU 作为一整组进行测试, 这导致一些性能良好的 CPU 无法通过测试, 生产部门对此颇有微词。作为质检部门的领导, 小 S 在不更改质检测试流程的前提下, 想出了这样一个主意: 如果能够把  $n$  台 CPU 恰当地分成连续的若干段, 使得每段 CPU 都能够通过成组测试, 就可以解决当下的问题。

现在, 小 S 已经知道了  $n$  台各自的性能表现  $P_1 \dots P_n$ , 两台 CPU 的 RPD 被定义为它们性能表现的差的绝对值。请你帮忙计算一下, 至少把这些 CPU 分成多少段, 才能使得每一段都能通过成组测试。

## 【输入格式】

每个测试点包含多组数据, 第一行一个整数  $T$  给出数据组数。

对于每组数据, 第一行三个整数  $n, m, k$ , 第二行  $n$  个整数  $P_1 \dots P_n$ 。

## 【输出格式】

对于每组数据, 输出一个整数表示答案。

样例输入	样例输出
2	2
5 1 49	1
8 2 1 7 9	
5 1 64	
8 2 1 7 9	

**【数据规模与约定】**

对于 20% 的数据， $1 \leq n \leq 10^2$ 。

对于 40% 的数据， $1 \leq n \leq 10^3$ 。

对于另外 10% 的数据， $k = 0$ 。

对于另外 10% 的数据， $0 \leq k \leq 1$ 。

对于另外 10% 的数据， $m = 1$ 。

对于另外 10% 的数据， $1 \leq m \leq 2$ 。

对于 90% 的数据， $0 \leq k \leq 10^{12}$ 。

对于 100% 的数据， $T \leq 12$ ,  $1 \leq n, m \leq 5 \cdot 10^5$ ,  $0 \leq k \leq 10^{18}$ ,  $0 \leq P_i \leq 2^{20}$ 。