

# NOI练习赛

fefb6944cc60c32f1ea5833326e27d61

中文题目名称	捡钱	配对	字符串匹配
英文题目名称	money	pair	kmp
输入文件名	money.in	pair.in	kmp.in
输出文件名	money.out	pair.out	kmp.out
每个测试点时限	5s	1s	1s
测试点数目	10	20	10
每个测试点分值	10	5	10
内存限制	256MB	256MB	256MB
是否有部分分	否	否	否
题目类型	传统	传统	传统

# 1 捡钱

## 1.1 题目描述

一开始地上没有钱，而每一时刻会发生下列事件之一：

- ①在某时刻，某个熊孩子往地上掉了一堆价值为 $A_i$ 的钱……
- ②在某时刻，一堆 $A_i$ 的钱被某个熊孩子捡走了……（如果本来没有这堆钱，就当什么都没有发生好了）
- ③在某时刻，你想知道地上一共还剩多少钱……
- ④在某时刻，你想知道地上剩下的钱堆的最小价值……（如果没有，请输出0）
- ⑤在某时刻，价值大于等于 $A_i$ 并且小于等于 $B_i$ 的钱全被某个熊孩子捡走了……
- ⑥在某时刻，价值大于等于 $A_i$ 并且小于等于 $B_i$ 的钱之中，价值最大的那一堆（如果有）被某个熊孩子捡走了……
- ⑦在某时刻，你想知道价值大于等于 $A_i$ 并且小于等于 $B_i$ 的钱中钱堆的最小价值……（没有符合题意的钱堆请输出0）
- ⑧在某时刻，你想知道价值大于等于 $A_i$ 并且小于等于 $B_i$ 的钱之中的第 $M_i$ 到 $N_i$ 贵重的这些钱之中的价值总和……（没有符合题意的钱堆请输出0）

数据保证，对于任一时刻，不存在两堆价值相同的钱。

附加说明：如果对⑧操作有细节疑问，可以看看这个具体解释：将价值大于等于 $A_i$ 并且小于等于 $B_i$ 的钱暂时取出记作 $Q$ ，然后对于 $Q$ 中的每一堆钱，如果其价值在 $Q$ 中的排名（降序排列）大于等于 $M_i$ 并且小于等于 $N_i$ ，则我们称这堆钱符合题意。所要输出的便是 $Q$ 中所有符合题意的钱堆的价值总和。当然如果没有符合题意的钱堆则输出0。

## 1.2 输入格式

输入文件为money.in。

第一行为一个整数 $N$ ，代表总的时间。

接下来 $N$ 行，每行若干个整数，表示每个时刻发生的事件。第 $i + 1$ 行表示第 $i$ 时刻发生的事件。第一个整数 $Type_i$ 在 $[1, 8]$ 以内，表示事件代号，如上所述。

- 如果 $Type_i \in \{1, 2\}$ ，接下来会有一个整数 $A_i$ ；
- 如果 $Type_i \in \{3, 4\}$ ，接下来不会有任何数；
- 如果 $Type_i \in \{5, 6, 7\}$ ，接下来会有两个整数 $A_i B_i$ ；
- 如果 $Type_i = 8$ ，接下来会有四个整数 $A_i B_i M_i N_i$ 。

测试数据中，输入数据均为随机生成的。它的意义是：对于每个 $i$ ， $Type_i$ 有一半概率为1，而其余的概率均等分给各其他合法的 $Type$ ；每个 $A_i B_i M_i N_i$ 均在合法范围内随机生成。

## 1.3 输出格式

输出文件为money.out。

对于每个1,2,5,6事件请输出一行OK，对于其他事件请输出询问的答案。

## 1.4 样例输入1

```
8
3
4
1 250
2 270
```

7 0 250  
8 250 250 2 3  
5 70 170  
6 120 700

### 1.5 样例输出1

0  
0  
OK  
OK  
250  
0  
OK  
OK

### 1.6 样例输入2

30  
1 50  
1 60  
1 70  
3  
4  
2 40  
3  
4  
2 60  
3  
4

2 50  
3  
4  
1 30  
1 80  
5 30 100  
3  
4  
1 40  
1 90  
1 60  
5 50 50  
6 40 80  
7 40 80  
8 40 80 2 3  
8 40 80 1 2  
1 60  
8 40 80 1 2  
7 60 90

## 1.7 样例输出2

OK  
OK  
OK  
180  
50  
OK  
180

50  
OK  
120  
50  
OK  
70  
70  
OK  
OK  
OK  
0  
0  
OK  
OK  
OK  
OK  
OK  
40  
0  
40  
OK  
100  
60

## 1.8 数据规模与约定

对于100%的数据， $1 \leq A_i, B_i, M_i, N_i \leq 1000000000$ 。

测试点编号	$Type_i$	$N$	其他特征
1	$\leq 3$	$\leq 500000$	$A_i \leq 1000000$
2	$\leq 3$	$\leq 500000$	无
3	$\leq 4$	$\leq 500000$	$A_i \leq 1000000$
4	$\leq 4$	$\leq 500000$	无
5	$\leq 6$	$\leq 100000$	无
6	$\leq 6$	$\leq 500000$	$A_i \leq 1000000$
7	$\leq 6$	$\leq 500000$	无
8	$\leq 8$	$\leq 100000$	无
9	$\leq 8$	$\leq 500000$	$A_i \leq 1000000$
10	$\leq 8$	$\leq 500000$	无

## 2 配对

### 2.1 题目描述

你在一个正方形网格中有 $2n$ 个珠子。这些珠子被涂成了 $n$ 种颜色，使得每种颜色恰好有2个珠子。这些珠子的坐标分别为 $(1, 0), (2, 0), \dots, (2n, 0)$ 。

你的任务是对于每种颜色画一条连接这两个珠子的路径：

- 每条路径应由水平和竖直的连接格点的线组成。
- 任意两条路径不能相交或触碰。
- 任意一条路径不能穿越直线 $y = 0$ 。每条路径只能在所连接的两个珠子处触碰直线 $y = 0$ （所以每个路径的第一条和最后一条线段一定是垂直的）。

给出珠子的排列，你需要算出可能的最小高度，或输出-1（如果无解）。一个方案的高度定义为所有路径中 $y$ 坐标的最大值与最小值之差。

一个例子：

```
red red blue yellow blue yellow
```

一个解是:

```
  +-----+
  |         |
red red blue yellow blue yellow
                        |
                        +-----+
```

在这种情况下最小的高度是2。

## 2.2 输入格式

输入文件为pair.in。

输入的第一行是数据组数 $T$ ，然后是 $T$ 组数据。每组数据的第一行是珠子个数 $n$ ，下一行包含空格隔开的 $2n$ 个英语单词，从左到右对应着每个珠子。每个颜色是长度不超过10的由小写英文字母组成的字符串。保证一共存在恰好 $n$ 种颜色，且每种颜色恰好出现2次。

## 2.3 输出格式

输出文件为pair.out。

对于每组数据，输出“Case #x: ”，其中 $x$ 是数据组数（从1开始），接下来是最小的高度。如果无解，输出-1。

## 2.4 样例输入

```
4
3
red red blue yellow blue yellow
```



3  
red blue yellow red blue yellow  
3  
red blue yellow blue yellow red  
3  
red red blue blue yellow yellow

## 2.5 样例输出

Case #1: 2  
Case #2: -1  
Case #3: 3  
Case #4: 1

## 2.6 数据规模与约定

对于所有数据， $T \leq 50$ 。  
20个测试点的 $n$ 的最大值分别为：  
2,4,6,8,10,  
12,14,16,18,20,  
50,100,150,200,250,  
300,350,400,450,500。

# 3 字符串匹配

## 3.1 题目描述

大家都知道斐波那契数列。现在把数换成字符串，定义斐波那契字符串列如下：

$F(0) = \text{"0"}$ ,

$F(1) = "1"$ ,

$F(n|n \geq 2) = F(n-1) + F(n-2)$ 。

其中加号定义为两个字符串的连接。

例如:  $F(2) = "10"$ ,  $F(3) = "101"$ ,  $F(4) = "10110"$ 。

给出一个01 串 $p$ 和自然数 $n$ , 求 $p$ 在 $F(n)$ 中出现的次数。保证答案可以用带符号64位整形表示。

### 3.2 输入格式

输入文件为kmp.in。

第一行一个数 $n$ 。

第二行为串 $p$ 。

### 3.3 输出格式

输出文件为kmp.out。

输出所求的出现次数。

### 3.4 样例输入

6

101

### 3.5 样例输出

4

### 3.6 样例解释

$F(6) = "1011010110110"$ , 其中出现了4 次“101”。

### 3.7 数据规模与约定

设 $m$ 为 $p$ 的长度:

对于30%的数据,  $n \leq 16$ 。

对于60%的数据,  $n \leq 25$ 。

另有20%的数据,  $m = 1$ 。

对于100%的数据,  $0 \leq n \leq 100, 1 \leq m \leq 100000$ 。