

# NOIP 模拟赛

ZZFLS

(请选手务必仔细阅读本页内容)

## 一、题目概况

|         |                    |          |            |            |
|---------|--------------------|----------|------------|------------|
| 中文题目名称  | 01 串               | 翻转匹配     | 建筑修缮       | 旅行商问题      |
| 子目录名    | bit                | flip     | repair     | travel     |
| 可执行文件名  | bit                | flip     | repair     | travel     |
| 输入文件名   | bit.in             | flip.in  | repair.in  | travel.in  |
| 输出文件名   | bit.out            | flip.out | repair.out | travel.out |
| 每个测试点时限 | 1000ms             | 1000ms   | 5000ms     | 2000ms     |
| 内存上限    | 512M               | 512 M    | 512M       | 512M       |
| 附加样例文件  | 有                  | 有        | 有          | 有          |
| 结果比较方式  | 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         | 传统         |

## 二、提交源程序程序名

|           |         |          |            |            |
|-----------|---------|----------|------------|------------|
| 对于 C++ 语言 | bit.cpp | flip.cpp | repair.cpp | travel.cpp |
|-----------|---------|----------|------------|------------|

## 三、优化开关

|           |                |
|-----------|----------------|
| 对于 C++ 语言 | -O2 -std=c++14 |
|-----------|----------------|

### 注意事项:

1. 文件名 (程序名和输入输出文件名) 必须使用英文小写;
2. C/C++ 中函数 main() 的返回类型必须是 int, 程序正常结束时的返回值必须是 0;
3. 保证各个题目的时间限制至少为标程运行时间的两倍;
4. 每道题目需单独建立对应子文件夹.

# 1 01 串

(bit.cpp)

## 1.1 问题描述

你有一个二进制串长度为  $N$ ，串内包含 0 和 1 两个数字。现在用一种特殊的算法对该串进行加密，加密方式是给定一个整数  $K$  满足  $1 \leq K \leq N$ 。对于该串内每个长度为  $K$  的区间，计算出该区间内数字的和，放进一个新序列里。新序列一共有  $N - K + 1$  项，第  $i$  项代表原序列中第  $i$  项到第  $i + K - 1$  项的和。

现在给你加密后的序列，请你求出原序列有多少种不同的可能性？为了避免答案过大，请你对 1000003 取模。

## 1.2 输入

第一行输入两个正整数  $N, K$ 。

其后一行  $N - K + 1$  个整数，代表序列  $A$ 。

## 1.3 输出

输出一行一个整数，代表可能的原序列的数量对 1000003 取模的值。

## 1.4 输入输出样例 1

### 1.4.1 输入样例

```
7 4
3 2 2 2
```

### 1.4.2 输出样例

```
3
```

## 1.5 样例解释

对样例一：

三种可能的情况是：“1011001”，“1101010”，“1110011”

## 1.6 约定和数据范围

对所有测试点数据，保证  $1 \leq N \leq 10^6, 1 \leq K \leq N$ 。

每个测试点的具体限制见下表：

| 子任务编号 |    | $n$                  | $k$                  |
|-------|----|----------------------|----------------------|
| 1     | 10 | $1 \leq N \leq 10$   | $1 \leq k \leq 3$    |
| 2     | 10 | $1 \leq N \leq 10$   | $1 \leq k \leq n$    |
| 3     | 20 | $1 \leq N \leq 1000$ | $1 \leq k \leq 10$   |
| 4     | 20 | $1 \leq N \leq 10^6$ | $1 \leq k \leq 20$   |
| 5     | 20 | $1 \leq N \leq 10^6$ | $1 \leq k \leq 3000$ |
| 6     | 20 | $1 \leq N \leq 10^6$ | $1 \leq k \leq n$    |

## 2 翻转匹配

(flip.cpp)

### 2.1 问题描述

Alice 正在玩一个翻转串的游戏。这个游戏有一个 S 串一个 T 串。两个串都是 01 串。Alice 每次操作可以把 S 串的一个子串翻转。例如”101100”，她选择”011” 翻转后得到”111000”。

Alice 希望进行最少次的操作，使得操作后的 S 串内**不包含** T 作为 S 的子串，你能否帮助 Alice 计算一下，最少需要多少次的翻转操作才能满足条件？如果无论怎么操作都不能满足条件，输出-1

### 2.2 输入

输入有两行，第一行一个 01 串，代表串 S，第二行一个 01 串，代表串 T。

### 2.3 输出

输出一行一个整数，表示 Alice 操作的最小次数，如果无解输出-1

### 2.4 输入输出样例 1

#### 2.4.1 输入样例

```
100110
10
```

#### 2.4.2 输出样例

```
2
```

### 2.5 输入输出样例 2

```
000
00
```

#### 2.5.1 输出样例

```
-1
```

### 2.6 约定和数据范围

对所有测试点数据，保证  $1 \leq |S| \leq 200000, 1 \leq |T| \leq 3$

每个测试点的具体限制见下表：

| 子任务编号 | 分值 | T 串限制                   |
|-------|----|-------------------------|
| 1     | 4  | $ T  = 1$               |
| 2     | 12 | $ T  = 2, T_1 \neq T_2$ |
| 3     | 16 | $ T  = 2$               |
| 4     | 20 | $ T  = 3, T_1 \neq T_3$ |
| 5     | 20 | $ T  = 3, T_1 \neq T_2$ |
| 6     | 28 | $ T  = 3$               |

### 3 建筑修缮

(repair.cpp)

#### 3.1 问题描述

富有的大建筑师 Alice 正在修缮她的建筑。她希望自己的建筑能呈现一个单峰的形状。具体来说: 把建筑的形状转化为一个长度为  $N$  的整数序列  $h$ , 存在一个  $p \in [1, N]$  满足  $h_1 \leq h_2 \leq \dots \leq h_p \geq \dots \geq h_{N-1} \geq h_N$

修缮的方式是, 每次选择三个位置  $(i, j, k)$ , 满足  $1 \leq i < j < k \leq N$ , 并且  $h_i > h_j < h_k$ 。然后将  $h_j$  增加 1, 该次操作的花费是  $h_i + h_j + h_k$ 。注意此处的  $h_j$  指增加前的  $h_j$

Alice 请你帮忙计算一下, 她最少需要花费多少代价能使得建筑呈现一个单峰的形状?

#### 3.2 输入

第一行输入一个整数  $N$ , 代表序列长度。

第二行  $N$  个整数, 代表  $h$  序列。

#### 3.3 输出

输出一行一个整数, 代表答案。由于答案过大, 请你输出答案对 1000003 取模的结果。

#### 3.4 输入输出样例 1

##### 3.4.1 输入样例

```
8
3 2 4 5 4 1 2 1
```

##### 3.4.2 输出样例

```
14
```

#### 3.5 样例解释

第一步选择 (2,6,7) 三个位置, 给  $h_6$  增加 1, 花费为 5, 变成 [3,2,4,5,4,2,2,1]

第二步选择 (1,2,5) 三个位置, 给  $h_2$  增加 1, 花费为 9, 变成 [3,3,4,5,4,2,2,1], 此时已经符合条件。

#### 3.6 约定和数据范围

对所有测试点数据, 保证  $3 \leq N \leq 10^6, 1 \leq h_i \leq 10^9$

每个测试点的具体限制见下表:

| 子任务编号 | 得分 | $n$                  | $h$  |
|-------|----|----------------------|--|
| 1     | 12 | $3 \leq N \leq 5000$ | $1 \leq h_i \leq 100$ , 且保证存在 $p$ 满足 $h_1 \geq h_2 \geq \dots \geq h_p \leq \dots \leq h_{N-1} \leq h_N$ |
| 2     | 12 | $3 \leq N \leq 5000$ | $1 \leq h_i \leq 100$  |
| 3     | 12 | $3 \leq N \leq 5000$ | $1 \leq h_i \leq 10^6$   |
| 4     | 12 | $3 \leq N \leq 5000$ | $1 \leq h_i \leq 10^9$   |
| 5     | 16 | $3 \leq N \leq 10^6$ | $1 \leq h_i \leq 100$  |
| 6     | 20 | $3 \leq N \leq 10^6$ | $1 \leq h_i \leq 10^6$   |
| 7     | 16 | $3 \leq N \leq 10^6$ | $1 \leq h_i \leq 10^9$   |

## 4 旅行商问题

(travel.cpp)

### 4.1 问题描述

一个交易员在  $T$  国做生意，该国家有  $N$  个城市， $N$  个城市之间有  $N-1$  条道路，每条道路连接两个城市，穿过每条道路需要一天的时间，可以通过这些道路从一个城市到达任意另一个城市。交易员起初在城市 1。

第  $i$  个城市可以获得  $p_i$  的利润，但是这个利润只能获得一次，每当该交易员到达一个城市，他可以选择获取或者不获取该城市的利润。并且他可以到达一个城市多次。

但是交易员的老板会随时监控他的收益，如果连续  $K$  天交易员没有获得收益，那么他将被解雇。我们想知道，交易员能获得的最大利润是多少？

### 4.2 输入

第一行，两个正整数  $N, K$

其后  $N-1$  行，每行输入包含两个以空格分隔的整数  $u_i$  和  $v_i$ , ( $1 \leq u_i, v_i \leq N, u_i \neq v_i$ ) 表示一条道路  
其后一行输出  $N$  个整数  $p_1, p_2, \dots, p_N$ ，代表每个城市的利润

### 4.3 输出

第一行，输出最大可能的总利润

第二行，输出  $M$ ，表示交易员选择获取利润的城市数量

第三行，输出  $M$  个整数， $x_1, x_2, \dots, x_M$ ，交易员将按照该路径依次获取收益，并且相邻两点之间默认走最短路径。注意  $x_1$  需要等于 1。如果有多种方案，输出其中任意一种即可。

### 4.4 输入输出样例 1

#### 4.4.1 输入样例

```
4 1
1 2
1 3
2 4
3 1 4 1
```

#### 4.4.2 输出样例

```
7
2
1 3
```



## 4.5 输入输出样例 2

### 4.5.1 输入样例

```
5 2
1 2
1 3
2 4
2 5
3 1 4 1 5
```

### 4.5.2 输出样例

```
14
5
1 4 5 2 3
```

## 4.6 约定和数据范围

对所有测试点数据, 保证  $2 \leq N \leq 200000, 1 \leq K \leq 3$ .

每个测试点的具体限制见下表:

| 子任务编号 | 分值 | $n$                    | $K$     |
|-------|----|------------------------|---------|
| 1     | 8  | $2 \leq N \leq 200000$ | $K = 1$ |
| 2     | 32 | $2 \leq N \leq 200$    | $K = 2$ |
| 3     | 12 | $2 \leq N \leq 2000$   | $K = 2$ |
| 4     | 16 | $2 \leq N \leq 200000$ | $K = 2$ |
| 5     | 16 | $2 \leq N \leq 2000$   | $K = 3$ |
| 6     | 20 | $2 \leq N \leq 200000$ | $K = 3$ |