

# 10.28 NOIP模拟测试

---

时间：2023年10月28日 8:00-11:30

## A.JJOII (joi.cpp/in/out)

---

2s,256MB

### 题目描述

---

定义有连续  $K$  个 J 和连续  $K$  个 0 和连续  $K$  个 I 组成的字符串为  $K$  阶 JOI 串。

比如，JJ00II 为 2 阶 JOI 串，**但是，注意要有顺序**，比如 00JJII 就不是 2 阶 JOI 串。

现在，给定一个长度为  $N$  的字符串  $S$ ，可以对它进行 3 种操作：

- 操作 1：删除  $S$  开头的字符
- 操作 2：删除  $S$  结尾的字符
- 操作 3：删除  $S$  除了开头和结尾之外的一个字符

我们要通过这些操作让  $S$  变为  $K$  阶 JOI 串。

但是，我们想让操作 3 尽可能的少。

所以我们想知道，变为  $K$  阶 JOI 串操作 3 最少需要进行多少次？

如果不能变为  $K$  阶 JOI 串，那么输出  $-1$ 。

### 输入格式

---

第一行两个整数  $N, K$  代表字符串长度和要构造的 JOI 串的阶数。

第二行  $N$  个字符代表字符串  $S$ 。

### 输出格式

---

一行一个整数代表操作 3 的最小进行次数。

如果不能变为  $K$  阶 JOI 串，那么输出  $-1$ 。

## 样例 #1

---

### 样例输入 #1

```
10 2
OJIJOIOIIIJ
```

### 样例输出 #1

```
2
```

## 样例 #2

---

### 样例输入 #2

```
9 3
JJJ000III
```

### 样例输出 #2

```
0
```

## 样例 #3

---

### 样例输入 #3

```
9 1
III000JJJ
```

### 样例输出 #3

```
-1
```

## 提示

### 样例 1 解释

1. 进行一次操作 1, 变为 JIJ0IOIIJ。
2. 进行一次操作 2, 变为 JIJ0IOII。
3. 进行一次操作 3, 删掉字符 2, 变为 JJ0IOII。
4. 进行一次操作 3, 删掉字符 4, 变为 JJ00II。

### 样例 2 解释

JJJ000III 已经是 3 阶 JOI 串了, 所以不需要进行操作。

### 样例 3 解释

III000JJJ 无法变为 1 阶 JOI 串, 无解。

### 数据规模与约定

本题采用捆绑测试。

- Subtask 1 (1 pts) :  $N \leq 21$ 。
- Subtask 2 (12 pts) :  $N \leq 3000$ 。
- Subtask 3 (87 pts) : 无特殊限制。

对于 100% 的数据:

- $3 \leq N \leq 2 \times 10^5$ 。
- $1 \leq K \leq \frac{N}{3}$ 。
- $S$  只包含 J, 0, I 且长度为  $N$ 。

## B. 贪婪的牛 (greedy.cpp/in/out)

1s, 128MB

### 题目描述

Farmer John 有  $M$  头奶牛, 为了方便, 编号为  $1, \dots, M$ 。这些奶牛平时都吃青草, 但是喜欢偶尔换换口味。Farmer John 一天烤了  $N$  个派请奶牛吃, 这  $N$  个派编号为  $1, \dots, N$ 。第  $i$  头奶

牛喜欢吃编号在  $[l_i, r_i]$  中的派（包括两端），并且没有两头奶牛喜欢吃相同范围的派。第  $i$  头奶牛有一个体重  $w_i$ ，这是一个在  $[1, 10^6]$  中的正整数。

Farmer John 可以选择一个奶牛序列  $c_1, c_2, \dots, c_K$ ，并让这些奶牛按这个顺序轮流吃派。不幸的是，这些奶牛不知道分享！当奶牛吃派时，她会把她喜欢吃的派都吃掉——也就是说，她会吃掉编号在  $[l_{c_i}, r_{c_i}]$  中所有剩余的派。Farmer John 想要避免当轮到一头奶牛吃派时，她所有喜欢的派在之前都被吃掉了这样尴尬的情况。因此，他想让你计算，要使奶牛按  $c_1, c_2, \dots, c_K$  的顺序吃派，轮到这头奶牛时她喜欢的派至少剩余一个的情况下，这些奶牛的最大可能体重（ $w_{c_1} + w_{c_2} + \dots + w_{c_K}$ ）是多少。

## 输入格式

第一行包含两个正整数  $N, M$ ；

接下来  $M$  行，每行三个正整数  $w_i, l_i, r_i$ 。

## 输出格式

输出对于一个合法的序列，最大可能的体重值。

## 样例 #1

### 样例输入 #1

```
2 2
100 1 2
100 1 1
```

### 样例输出 #1

```
200
```

## 提示

### 样例解释

在这个样例中，如果奶牛 1 先吃，那么奶牛 2 就吃不到派了。然而，先让奶牛 2 吃，然后奶牛 1 只吃编号为 2 的派，仍可以满足条件。

对于全部数据,  $1 \leq N \leq 300, 1 \leq M \leq \frac{N(N-1)}{2}, 1 \leq l_i, r_i \leq N, 1 \leq w_i \leq 10^6$ 。

### 数据范围

对于测试点 2 – 5, 满足  $N \leq 50, M \leq 20$ ;

对于测试点 6 – 9, 满足  $N \leq 50$ 。

## C.Intervals (intervals.cpp/in/out)

2s,1024MB

### 题目描述

给定  $m$  条规则形如  $(l_i, r_i, a_i)$ , 对于一个 01 串, 其分数的定义是: 对于第  $i$  条规则, 若该串在  $[l_i, r_i]$  中至少有一个 1, 则该串的分数增加  $a_i$ 。

你需要求出长度为  $n$  的 01 串中的最大分数。

$1 \leq n, m \leq 2 \times 10^5, |a_i| \leq 10^9$ 。

### 输入格式

输入按照以下格式:

$N \ M$

$l_1 \ r_1 \ a_1$

$l_2 \ r_2 \ a_2$

:

$l_M \ r_M \ a_M$

### 输出格式

输出 01 字符串分数的最大值。

### 样例 #1

## 样例输入 #1

```
5 3
1 3 10
2 4 -10
3 5 10
```

## 样例输出 #1

```
20
```

## 样例 #2

---

### 样例输入 #2

```
3 4
1 3 100
1 1 -10
2 2 -20
3 3 -30
```

### 样例输出 #2

```
90
```

## 样例 #3

---

### 样例输入 #3

```
1 1
1 1 -10
```

### 样例输出 #3

```
0
```

## 样例 #4

---

### 样例输入 #4

```
1 5
1 1 1000000000
1 1 1000000000
1 1 1000000000
1 1 1000000000
1 1 1000000000
1 1 1000000000
```

### 样例输出 #4

```
5000000000
```

## 样例 #5

---

### 样例输入 #5

```
6 8
5 5 3
1 1 10
1 6 -8
3 6 5
3 4 9
5 5 -2
1 3 -6
4 6 -7
```

### 样例输出 #5

```
10
```

## 提示

---

### 数据范围

- 输入的数字全部为整数

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq M \leq 2 \times 10^5$
- $1 \leq l_i \leq r_i \leq N$
- $|a_i| \leq 10^9$

### 样例#1 解释

字符串为 10001 时，分数为最大值：  $a_1 + a_3 = 10 + 10 = 20$ 。

### 样例#2 解释

字符串为 100 时，分数为最大值：  $a_1 + a_2 = 100 + (-10) = 90$ 。

### 样例#3 解释

字符串为 0 时，分数为最大值 0。

### 样例#4 解释

答案超出了32位整数。

### 样例#5 解释

字符串为 101000 时，分数为最大值  $a_2 + a_3 + a_4 + a_5 + a_7 = 10 + (-8) + 5 + 9 + (-6) = 10$ 。

## D.道路中断 (disruption.cpp/in/out)

1s,128MB

### 题目描述

Farmer John自豪于他所经营的交通发达的农场。这个农场是由  $N$  块牧场 ( $2 \leq N \leq 50,000$ ) 组成的， $N - 1$  条双向道路将它们连接起来，每一条道路的都为一单位长度。Farmer John注意到，从任何一块牧场到另一块牧场，都能通过一组合适的道路到达。

尽管FJ的农场现在是连通的，他担心如果有一条道路被阻断会发生什么，因为这事实上会将他的农场分为两个不相交的牧场集合，奶牛们只能够在每一个集合内移动但不能在集合间移动。于是FJ又建造了  $M$  条额外的双向道路 ( $1 \leq M \leq 50,000$ )，每一条的长度都是一个至多为  $10^9$  的正整数。奶牛们仍然可以使用原有的道路进行移动，除非其中的某些被阻断了。



如果某条原有的道路被阻断了，农场就会被分为两块不相交的区域，那么FJ就会从他的额外修建的道路中选择一条能够重建这两块区域的连通性的，取代原来那条，从而奶牛们又可以从任何一块牧场去往另一块牧场。

对于农场上每一条原有的道路，帮助FJ选出最短的替代用的道路。

## 输入格式

输入的第一行包含 $N$ 和 $M$ 。接下来的 $N - 1$ 行，每行用整数 $p$ 和 $q$ 描述了一条原有的道路，其中 $p \neq q$ 是这条道路连接的两块牧场（在 $1 \dots N$ 范围内）。剩下的 $M$ 行，每行用三个整数 $p$ 、 $q$ 和 $r$ 描述了一条额外的道路，其中 $r$ 是这条道路的长度。任何两块牧场之间至多只有一条道路。

## 输出格式

对原有的 $N - 1$ 条道路的每一条，按照它们在输入中出现的顺序，输出如果这条道路被阻断的话，能够重新连接农场的最短的替代用道路的长度。如果不存在合适的替代用的道路，输出-1。

## 样例 #1

### 样例输入 #1

```
6 3
1 2
1 3
4 1
4 5
6 5
2 3 7
3 6 8
6 4 5
```

### 样例输出 #1

```
7
7
8
5
5
```