

# NOIP模拟赛 by Hercier

---

## NOIP模拟赛 by Hercier

特殊提示

格式及说明

刺客信条 (AC)

输入格式

输出格式

样例1

输入

输出

数据范围

黑暗之魂(darksoul)

题目描述

输入格式

输出格式

样例1

输入

输出

数据范围

传送门 (portal)

输入格式

输出格式

样例1

输入

输出

## 特殊提示

---

本题读入数据较大，如有需求，可以使用下面的函数进行读入优化。

```
const int MAXSIZE = 1 << 22;
inline char gc() {
    static char In[MAXSIZE], *at = In, *en = In;
    if (at == en) {
        en = (at = In) + fread(In, 1, MAXSIZE, stdin);
    }
    return at == en ? EOF : *at++;
}
inline long long gt() {
    char c;
    while (c = gc(), !(c >= '0' && c <= '9') && c != '-') {}
    bool f = c == '-';
```

```

    long long x = f ? 0 : c - '0';
    for (c = gc(); c >= '0' && c <= '9'; c = gc()) {
        x = x * 10 + c - '0';
    }
    return f ? -x : x;
}
int main(){
    freopen("*.in", "r", stdin);
    freopen("*.out", "w", stdout);
    int n;
    n=gt();
    long long a;
    a=gt();
}
//powered by skqliao

```

注意，`gt()` 只能进行文件输入。

## 格式及说明

题目名称后的括号内是程序名称。如a(b),则程序名应为b.cpp/.c/.pas,输入文件为b.in,输出文件为b.out.文件名区分大小写。如无特殊说明，空间限制为512MB，时间限制为1000ms，栈空间大小为512MB,代码文件大小应小于50KB，比较方式默认为：逐行比较，忽略多余的空格和制表符。

## 刺客信条（AC）

故事发生在1486年的意大利，*Ezio* 原本只是一个文艺复兴时期的贵族，后来因为家族成员受到圣殿骑士的杀害，决心成为一名刺客。最终，凭借着他的努力和出众的天赋，成为了杰出的刺客大师。刺客组织在他的带领下，为被剥削的平民声张正义，赶跑了原本统治意大利的圣殿骑士首领-教皇亚历山大六世。在他的一生中，经历了无数次惊心动魄、扣人心弦的探险和刺杀。

这次的故事就是他暗杀一位作恶多端的红衣主教。红衣主教可以吸取他周围人的生命力量，而他的红衣教徒也拥有这个力量。红衣主教的家是一个 $x \times y$ 的长方形房间，也就是说，他的家的四个角坐标分别为 $(0, 0)$   $(x, 0)$   $(0, y)$   $(x, y)$ 。教堂的门在 $(0, 0)$ ,而红衣主教就在 $(x, y)$ 的卧室休息。他的家中还有 $n$ 个守护着他的红衣教徒，站在 $(a_i, b_i)$ 。*Ezio*想要趁主教休息时，**从门进入**潜入到他的卧室刺杀他，因为主教休息时会脱下红衣，这样吸取生命的力量就消失了。可是守卫他的红衣教徒依然很危险，离红衣教徒太近就会被吸取生命。因此，*Ezio*想知道，在能刺杀主教的前提下，从门到他的卧室的路上，他最远和离他最近的红衣教徒保持多远的距离。**注意：教徒都在房间里。**

## 输入格式

第一行三个整数 $x, y, n$ 。之后 $n$ 行，每行两个整数 $a_i, b_i$ ,意义见题目描述。

## 输出格式

一行一个数 $D$ ，表示*Ezio*能保持的最大距离，保留两位小数。

## 样例1

## 输入

```
10 20 2
3 3
6 14
```

## 输出

```
3.00
```

样例说明

贴着墙走

## 数据范围

对 10%的数据  $n \leq 10$  对 30%的数据  $n \leq 100$  对 100%的数据  $n \leq 2000$  保证输入合法,  $x, y \in [1, 10^6]$

## 黑暗之魂(darksoul)

### 题目描述

oi\_juruo热爱一款名叫黑暗之魂的游戏。在这个游戏中玩家要操纵一名有 $hp$ 点生命值的无火的余灰在一张地图中探险。地图中有 $n$ 个篝火（也就是存档点）。在篝火处休息可以将生命值恢复满。每个篝火都会向其他篝火的其中之一连有一条通道（显然，通道是双向的），这些篝火之间都相互可达。也就是说，这是一张 $n$ 个点， $n$ 条边的无向连通图。每条通道里都有一些怪物，经过oi\_juruo的分析，他得到了每条边的怪物会对他造成的伤害值 $a_i$ 。为了向oier们表演他高超的游戏技巧，他要从任意一个篝火跑到任意另一个篝火而不在之间的篝火休息，在此期间，他会和他经过的通道中的怪物战斗并损失 $a_i$ 的生命值。现在oi\_juruo想知道，他的生命值 $hp$ 至少为多少，才能完成任意一条旅途。oi\_juruo并不傻，他会走最安全的路。**本题时限为3000ms**

### 输入格式

第一行一个整数 $n$ 。之后 $n$ 行，每行三个整数 $u_i, v_i, a_i$ ，表示有一条从 $u_i$ 连向 $v_i$ ，怪物伤害值为 $a_i$ 的通道。

### 输出格式

一行一个数 $hp$ ，表示无火的余灰的最小生命值。

### 样例1

#### 输入

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

#### 输出

样例说明

从2到5的路最危险， $2 \rightarrow 1 \rightarrow 4 \rightarrow 5$ 受到了7点伤害，所以需要8点生命值。

## 数据范围

| 数据点编号     | n              | 性质      |
|-----------|----------------|---------|
| 1(5%)     | 10             | 无自环，无重边 |
| 2,3(10%)  | $\leq 2000$    | 无自环，无重边 |
| 4,5(10%)  | $\leq 7000$    | 无自环，无重边 |
| 6(5%)     | $\leq 1000000$ | 有自环，无重边 |
| 7(5%)     | $\leq 1000000$ | 无自环，有重边 |
| 8~20(65%) | $\leq 1000000$ | 无自环，无重边 |

保证输入合法， $\forall i \in [1, n], a_i \in [1, 1e9]$

## 传送门（portal）

8102年，*Normalgod*在GLaDOS的帮助下，研制出了传送枪。但GLaDOS想把传送枪据为己有，于是把*Normalgod*扔进了一间实验室。这间实验室是一棵有n个节点的树。现在*Normalgod*在一号节点，出口也在一号节点，但为了打开它，必须经过每一个节点按下每个节点的开关，出口才能打开。GLaDOS为了杀死*Normalgod*，开始在实验室里释放毒气，因此*Normalgod*必须尽快逃出这间实验室。

当然，*Normalgod*手中的传送枪是可以使用的。传送枪可以发射出**两个**颜色不同的传送门。*Normalgod*可以从其中一个传送到另一个。尽管传送枪可以在视野范围内的任何一个经过特殊处理的表面打开一扇传送门，但这间实验室的设计使得*Normalgod*只能在他所处的房间内打开一个传送门。在已经存在了一个同颜色的传送门时，打开新的传送门会使与它同颜色的旧门消失。传送和打开传送门所需时间为0。

显然，利用传送枪会让*Normalgod*更快解决谜题，可*Normalgod*死在了按下最后一个按钮的路上。尽管如此，GLaDOS还是很想知道到底*Normalgod*最快能用多久逃出去，这对她的实验室设计方法论有重要的指导作用。作为GLaDOS的算法模块，你要完成这个任务。**本题时限为2000ms**

## 输入格式

第一行一个整数n。之后n-1行，每行三个整数 $u_i, v_i, a_i$ ，表示有一条从 $u_i$ 连向 $v_i$ ，花费时间为 $a_i$ 的通道。

## 输出格式

一行一个数T，表示最小的脱逃时间。

## 样例1

### 输入

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

输出

13

样例说明

1→open1→5→open2→use(1)→2→3→open2→use(1)→2→4→open2→use(1)→exit

数据范围

| 数据点编号      | n         | 性质         |
|------------|-----------|------------|
| 1,2(10%)   | ≤5(4,5)   |            |
| 3,4,5(15%) | ≤ 13      |            |
| 6~9(20%)   | ≤ 100     | 叶子节点数量≤ 12 |
| 10~14(25%) | ≤ 1000000 | 叶子节点数量≤ 12 |
| 15~20(30%) | ≤ 1000000 |            |

保证输入合法,  $\forall i \in [1, n), a_i \in [1, 1e9]$