

# GDSY-PION 模拟赛

	草莓列车	草莓路径	草莓城市	草莓之歌
输入文件名	train.in	path.in	city.in	easy.in
输出文件名	train.out	path.out	city.out	easy.out
源程序文件名	train.cpp	path.cpp	city.cpp	easy.cpp
时间限制	1s	1s	1s	3s
空间限制	512MB	512MB	512MB	512MB
题目类型	传统	传统	传统	传统
结果比较方式	全文比较	全文比较	全文比较	全文比较
是否捆绑测试	是	是	否	是

## Notice

- 请建立子文件夹。
- 题目难度请选手自行辨别。
- 编译命令：`-lm -O2 -std=c++14 -w1,--stack=998244353`

# 草莓列车(train)

## 题目背景

开草莓列车一定要快，落地了草莓就没了。

## 题目描述

给定一个序列  $\{a_n\}$ ，有  $m$  次操作，形如 `l r v`，表示将  $[l, r]$  的每个  $a_i$  变为  $\max(a_i, v)$ 。

求最终的序列。

## 输入格式

从 `train.in` 中读入数据。

为了避免过量的输入，此题使用特殊的读入格式。

第一行三个正整数  $n, m, type$  表示序列长度、操作次数和数据类型。

当  $type = 1$  时，请手动固定左端点  $l = 1$ 。

第二行包含  $n$  个数，表示初始的序列。

最后一行输入  $x_0, seed$ ，使用示例如下：

```
1 namespace Maker{
2     unsigned int x0,seed;
3     void init() {scanf("%u%u",&x0,&seed);}
4     inline unsigned int getnum(){
5         x0=(x0<<3)^x0;
6         x0=((x0>>5)+seed)^x0;
7         return x0;
8     }
9 }
10 int n,m,typ;
11 int main(){
12     scanf("%d%d%d",&n,&m,&typ);
13     // input n numbers as a[1...n]
14     Maker::init();
15     for(int i=1; i<=m; ++i){
16         int l=Maker::getnum()%n+1,r=Maker::getnum()%n+1;
17         unsigned int v=Maker::getnum();
18         if(l>r) swap(l,r);
19         if(typ==1) l=1;
20         // do something
21     }
22     // output n numbers of a[1...n]
23 }
```

## 输出格式

输出到 `train.out` 中。

一行  $n$  个整数，表示序列的每个数。

# 输入输出样例

## 样例输入 1

```
1 5 1 0
2 1 2 3 4 5
3 233 666
```

## 样例输出 1

```
1 1 94747 94747 94747 5
```

## 样例输入 2

```
1 5 1 1
2 1 2 3 4 5
3 456 789
```

## 样例输出 2

```
1 205550 205550 205550 4 5
```

样例解释 1：第一个操作为 2 4 94747。

样例解释 2：第一个操作为 1 3 205550。

# 数据范围

对于所有数据： $a_i \leq 10^5$ ，有且仅有子任务 4 的  $type = 1$ 。

子任务编号	限制	分值
1	$n, m \leq 5000$	20
2	$n, m \leq 10^5$	20
3	$n \leq 10^5, m \leq 10^6$	20
4	$n \leq 10^5, m \leq 10^7$	20
5	$n \leq 10^5, m \leq 10^7$	20

# 草莓路径(path)

## 题目描述

Madeline 有一张  $n$  个点  $m$  条边的无向联通图（可能存在重边、自环）。对于一条连接  $u_i$  和  $v_i$  的边，这条边上有  $w_i$  个草莓。定义一条路径的草莓值为这条路径的所有边上的草莓数量的异或和。

Madeline 最终能吃到草莓值这么多的草莓，所以请你找出草莓值最大的路径，输出这个值。注意路径的起点和终点可以相同。

## 输入格式

从 `path.in` 中读入数据。

第一行两个正整数  $n$  和  $m$ 。

第二行到第  $m + 1$  行每行 3 个整数  $u_i, v_i, w_i$ 。

## 输出格式

输出到 `path.out` 中。

一行共一个整数，表示草莓值的最大可能值。

## 输入输出样例

### 输入样例 1

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

### 输出样例 1

1	7
---	---

## 数据范围

对于所有数据： $n, m \leq 10^5, w_i < 10^{18}$ 。

子任务编号	限制	分值
1	$n \leq 10, m \leq 20$	15
2	$n, m \leq 3000$	20
3	$n, m \leq 7000$	15
4	无特殊限制	50

# 草莓城市(city)

## 题目描述

草莓城是一个四个角坐标分别为  $(0, 0)$ ,  $(W, 0)$ ,  $(0, H)$ ,  $(W, H)$  的矩形，其中有  $k$  个草莓，草莓所在的点都是整点。现在要给每个草莓建一个大棚，满足大棚都处在城市内，且互不相交（指被多个大棚覆盖的区域面积为零）。要求每个大棚的形状为等腰直角三角形，对应草莓处于斜边的中点，且斜边与一条坐标轴平行、所有三角形的斜边长度相等。

请你设计一个方案使得斜边的长度最大。

## 输入格式

从 `city.in` 中读入数据。

第一行三个整数  $W, H, k$ 。

接下来  $k$  行第  $i$  行两个整数  $x_i, y_i$  表示第  $i$  个草莓的坐标  $(x_i, y_i)$ 。草莓的坐标可以重复。

## 输出格式

输出到 `city.out` 中。

输出一行包含一个实数表示最大的斜边长度，要求恰好四舍五入到小数点后 2 位。保证答案是稳定的，即假设答案是  $ans$ ，则  $ans \pm \epsilon$  四舍五入的结果和  $ans$  一样。

## 输入输出样例

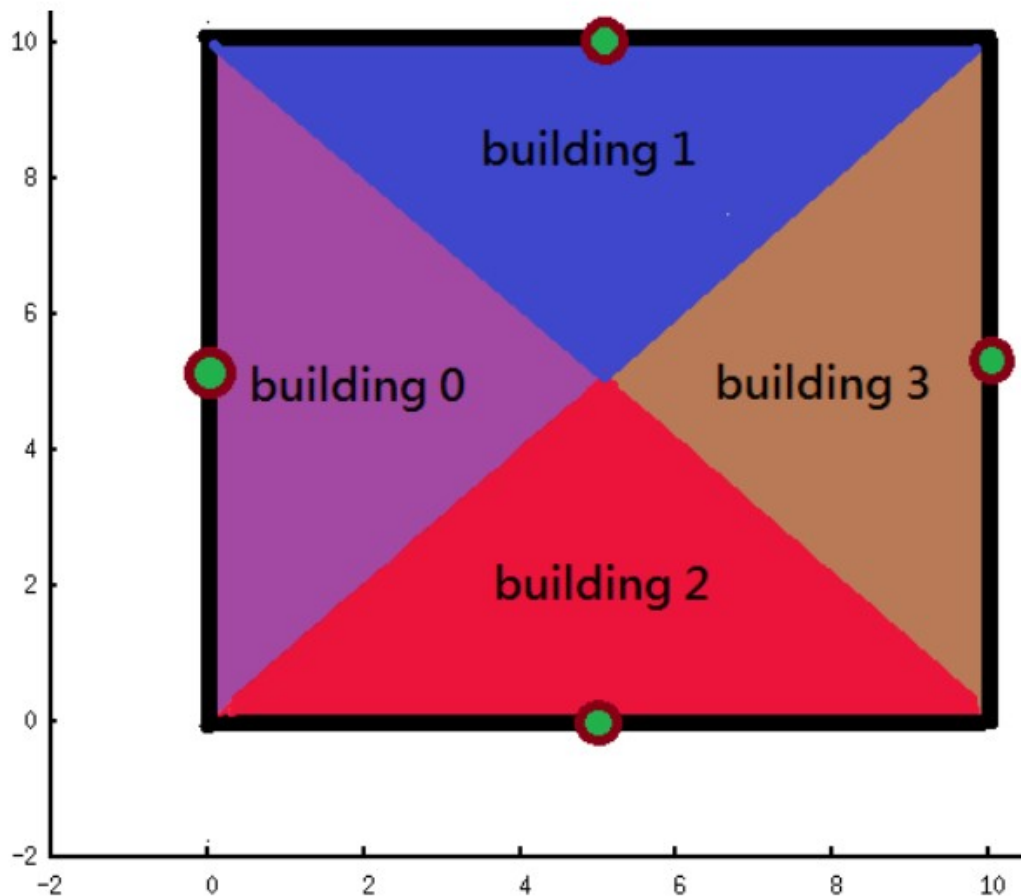
### 输入样例 1

```
1 | 100 100 4
2 | 5 0
3 | 5 10
4 | 0 5
5 | 10 5
```

### 输出样例 1

```
1 | 10.00
```

### 样例解释 1



## 输入样例 2

```
1 | 2 2 1
2 | 1 1
```

## 输出样例 2

```
1 | 2.00
```

## 数据范围

对于所有数据： $0 \leq W, H \leq 10^9, 1 \leq k \leq 200, 0 \leq x_i \leq W, 0 \leq y_i \leq H$ 。

对于 20% 的数据满足  $k \leq 4$ 。

对于 50% 的数据满足  $k \leq 50$ 。

# 草莓之歌(easy)

## 题目描述

在草莓国，每年都要举办演唱草莓歌的大型活动。

舞台上， $2N$  名演出者排成一排。草莓歌有两个声部：红草莓声部和金草莓声部。每一名演出者只能唱一种声部。这个信息用一个字符串  $S$  给出。具体地，对于从舞台的左边起第  $i$  名演出者，如果字符串  $S$  中第  $i$  个字符为 A，则这名演出者唱红草莓声部；如果为 B，则唱金草莓声部。保证字符串  $S$  中恰有  $N$  个 A 和 B。

现在开始，演出者们要唱  $K$  首草莓歌。然而因为草莓歌十分消耗体力，所以每名演出者只能恰好唱一首草莓歌。并且为了表示对草莓的崇拜，对于每首草莓歌都应满足以下条件：

- 至少一名演出者唱这首歌
- 唱红草莓声部的演出者的数量等于唱金草莓声部的演出者的数量
- 如果只考虑唱这首歌的演出者，则每名唱红草莓声部的演出者在舞台上都站在所有唱金草莓声部的演出者的左边

Madeline 导演要找到一种给每名演出者分配歌曲的方式，但目前可能并不存在一种合法的方案。

但是 Madeline 会使用名为 Wave Dash 的魔法。每使用一次魔法，Madeline 可以选择两名相邻的演出者，将他们交换。请你求出最少使用几次魔法，使得存在一种合法的分配方案。

## 输入格式

从 `easy.in` 中读入数据。

第一行两个整数  $N$  和  $K$ 。

第二行一个长度为  $2N$  的字符串  $S$ 。

## 输出格式

输出到 `easy.out` 中。

一行一个整数，代表答案。

## 输入输出样例

### 输入样例 1

```
1 5 3
2 AABABABBAB
```

### 输出样例 1

```
1 0
```

## 输入样例 2

```
1 5 3
2 AABABABBAB
```

## 输出样例 2

```
1 0
```

## 输出样例 3

```
1 3 1
2 BBAAAA
```

## 输出样例 3

```
1 9
```

## 输入样例 4

```
1 10 3
2 ABABBBBABBBABABABAAAA
```

## 输出样例 4

```
1 37
```

## 数据范围

对于所有输入数据，满足：

- $1 \leq N \leq 10^6$
- $1 \leq K \leq N$
- $S$  中包含  $N$  个字符 A 和  $N$  个字符 B

子任务编号	附加限制	分值
1	$N \leq 10$	16
2	$N \leq 500$	24
3	$N \leq 5000$	21
4	$N \leq 10^5$	26
5	无附加限制	13



