

# 2<sup>10</sup> 模拟赛

时间：10 月 24 日 8:00 – 12:30

题目名称	生物与非生物	次氯酸	玛恩纳	卡利贝尔
题目类型	传统型	传统型	传统型	传统型
可执行文件名	toad	hclo	mlynar	caliber
输入文件名	toad.in	hclo.in	mlynar.in	caliber.in
输出文件名	toad.out	hclo.out	mlynar.out	caliber.out
每个测试点时限	1.0 秒	3.0 秒	1.0 秒	1.0 秒
内存限制	512 MB	512 MB	512 MB	512 MB
提交源程序文件名	toad.cpp	hclo.cpp	mlynar.cpp	caliber.cpp
子任务/测试点数目	4	5	7	6
是否等分	否	否	否	否

编译选项： `-lm -O2 -std=c++17`

## 注意事项

1. C++ 中函数 `main` 的返回值类型必须是 `int`，程序正常返回时的返回值必须是 0。
2. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
3. 选手提交的程序代码文件请建立子目录，并在子目录外也保存一份代码。
4. 若无特殊说明，输入文件与输出文件中同一行的相邻整数均使用一个空格分隔。
5. 选手提交的程序源文件必须不大于 100 KB。
6. 程序可使用的栈空间内存限制与题目的内存限制一致。
7. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
8. 评测时采用的机器配置为 Intel(R) Core(TM) i5-10500@3.10GHz，内存 8 GiB。上述时限以此配置为准。
9. 评测在 Windows 10 下进行，使用 LemonLime 进行评测。
10. 题目不保证按难度排序。

# 生物与非生物 (toad)

## 题目背景

一只癞蛤蟆，整天呱呱呱。

除了赖赖赖，就是哈哈哈。

## 题目描述

Toad 定义一个 Legal sequence 如下：

- 空串是 Legal sequence。
- 如果  $A$  是 Legal sequence，那么  $(a)$ ,  $[a]$ ,  $\{a\}$  都是 Legal sequence。
- 如果  $A, B$  都是 Legal sequence，那么序列  $AB$  也是 Legal sequence。

Toad 现在发现了一个缺失的 Legal sequence，缺失的部分用 `?` 表示。

现在 Toad 请你求出有多少种可能的 Legal sequence。

## 输入格式

从文件 `toad.in` 中读入数据。

输入第一行为一个正整数  $n$ ，表示这个缺失的 Legal sequence。

第二行为  $n$  个字符的字符串  $S$ ，每个字符可能为 `?`, `{`, `}`, `[`, `]`, `(`, `)` 中的任何字符。

## 输出格式

输出到文件 `toad.out` 中。

输出一行一个整数，表示答案，由于答案可能很大，请对 998244353 取模。

## 输入输出样例

### 样例输入 #1

```
1 10
2 (?([?])???
```

### 样例输出 #1

```
1 3
```

### 样例输入 #2

```
1 40
2 ?????????????????????????????????????????
```

### 样例输出 #2

```
1 294115477
```

样例组 #3

见下发文件中的 toad/toad3.in, toad/toad3.out。

该组样例满足 Subtask #4 的性质。

样例解释 #1

所有可能的情况： `{{([O])}}` `O([O]{})` `([[]]{})`。

所以答案为 3。

数据范围与约定

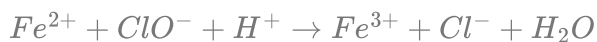
对于 100% 的数据，保证  $2 \leq n \leq 500, \forall 1 \leq i \leq n, S_i \in \{[, ], (, ), \{, \}, ?\}$ 。

子任务编号	$n$	特殊性质	分值
Subtask #1	$\leq 16$	无	10
Subtask #2	$\leq 50$	无	20
Subtask #3	$\leq 500$	$\forall 1 \leq i \leq n, S_i = ?$	30
Subtask #4	$\leq 500$	无	40

# 次氯酸（hclo）

## 题目背景

酸性条件下，次氯酸根将二价铁离子氧化为三价铁离子的离子方程式如下：



## 题目描述

给定一个  $n$  个节点的二叉树，编号 1 到  $n$ ，根为 1，对于任意一个非根节点  $i$ ， $i$  的父亲为  $\lfloor \frac{i}{2} \rfloor$ ，每一个节点  $j$  都有一个价值  $V_j$  和权重  $W_j$ 。

接下来有  $q$  次询问，每次询问给出两个正整数  $v, L$ ，请你在  $v$  和  $v$  的祖先中选出若干个（可以不选）节点，使得选出的节点的权重不超过  $L$ 。

次氯酸想知道最大能选出的价值。

## 输入格式

从文件 `hclo.in` 中读入数据。

第一行输入一个正整数  $n$ ，表示节点数。

接下来  $n$  行每行两个整数  $V_j, W_j$ ，含义如上文所述。

接下来一行一个整数  $q$ ，表示询问数。

接下来  $q$  行每行两个正整数  $v, L$ ，表示一组询问。

## 输出格式

输出到文件 `hclo.out` 中。

输出  $q$  行，每行一个整数表示答案。

## 输入输出样例

### 样例输入 #1

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

### 样例输出 #1

1	0
2	3
3	3

### 样例组 #2

见下发文件中的 `hclo/hclo2.in`, `hclo/hclo2.out`。

该组样例满足 Subtask #2 的限制。

**样例组 #3**

见下发文件中的 `hclo/hclo3.in`, `hclo/hclo3.out`。

该组样例满足 Subtask #3 的限制。

**样例组 #4**

见下发文件中的 `hclo/hclo4.in`, `hclo/hclo4.out`。

该组样例满足 Subtask #4 的限制。

**数据范围与约定**

对于 100% 的数据，保证  $1 \leq n < 2^{18}$ ,  $1 \leq q \leq 10^5$ ,  $1 \leq V_j, W_j, L \leq 10^5$ 。

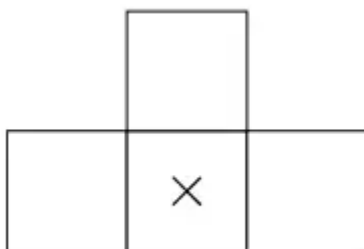
子任务编号	$n$	$q$	$L$	分值
Subtask #1	$< 2^{18}$	$\leq 10^5$	$= 1$	5
Subtask #2	$< 2^{10}$	$\leq 10^3$	$\leq 10^3$	10
Subtask #3	$< 2^{10}$	$\leq 10^4$	$\leq 10^5$	15
Subtask #4	$< 2^{16}$	$\leq 5 \times 10^4$	$\leq 10^5$	30
Subtask #5	$< 2^{18}$	$\leq 10^5$	$\leq 10^5$	40

# 玛恩纳 (mlynar)

## 题目背景

玛恩纳~

## 题目描述



如果你玩过明日方舟，你应该见过以上图形。

我们称它为一个 **M 形四格拼板**。其中心被标记为  $\times$ 。

玛恩纳画了一个  $n$  行  $m$  列的长方形网格。行从 0 至  $m - 1$  编号，列从 0 至  $n - 1$  编号。他将网格中的一些格子标记为**特殊格子**。

然后，他想要将 M 形四格拼板按如下规则摆放：

- 特殊格子的数量与 M 形四格拼板的数量相同，每个 M 形四格拼板的中心在网格上的位置必须是特殊格子。
- M 形四格拼板之间不能有重叠部分。
- 所有拼板的部分均在网格内。

注意，M 形四格拼板有四种摆放方式： $\perp$   $\top$   $\lrcorner$   $\ulcorner$ 。

如果方案不存在，输出 **No**，否则请找出一种方案使得被拼板覆盖的数总和最大，求出这个最大值。

## 输入格式

从文件 *mlynar.in* 中读入数据。

输入第一行为两个正整数  $m, n$  分别表示**行数和列数**。

接下来  $m$  行，每行  $n$  个整数，第  $i$  行第  $j$  个数（从 0 开始编号）表示方格中第  $i$  行第  $j$  列的数  $a_{i,j}$ 。

接下来一行，一个整数  $k$ ，表示特殊格子的数量。

接下来  $k$  行，每行两个整数  $r_i, c_i$ ，表示第  $i$  个被标记的特殊格子的位置。

## 输出格式

输出到文件 *mlynar.out* 中。

如果有方案，输出可能的被覆盖的格子内数总和的最大值，否则输出 **No**。

# 输入输出样例

## 样例输入 #1

1	5 6
2	7 3 8 1 0 9
3	4 6 2 5 8 3
4	1 9 7 3 9 5
5	2 6 8 4 5 7
6	3 8 2 7 3 6
7	3
8	1 1
9	2 2
10	3 4

## 样例输出 #1

1	67
---	----

## 样例输入 #2

1	5 6
2	7 3 8 1 0 9
3	4 6 2 5 8 3
4	1 9 7 3 9 5
5	2 6 8 4 5 7
6	3 8 2 7 3 6
7	3
8	1 1
9	2 2
10	3 3

## 样例输出 #2

1	No
---	----

## 样例组 #3

见下发文件中的 `mlynar/mlynar3.in`, `mlynar/mlynar3.out`。

该组样例满足 Subtask #4 的限制。

## 样例组 #4

见下发文件中的 `mlynar/mlynar4.in`, `mlynar/mlynar4.out`。

该组样例满足 Subtask #6 的限制。

# 数据范围与约定

对于 100% 的数据，保证  $1 \leq k \leq nm \leq 10^6$ ,  $a_{i,j} \in [0, 10^3]$ ,  $r_i \in [0, m)$ ,  $c_i \in [0, n)$ 。

子任务编号	$k$	特殊性质	分值
Subtask #1	$\leq 10^3$	A	5
Subtask #2	$\leq 10^3$	B	10
Subtask #3	$\leq 10^3$	C	10
Subtask #4	$\leq nm$	D	10
Subtask #5	$\leq 10$	无	15
Subtask #6	$\leq 10^3$	无	20
Subtask #7	$\leq nm$	无	30

特殊性质 A: 保证对于所有  $i \neq j$ ,  $\max(|r_i - r_j|, |c_i - c_j|) > 2$ 。

特殊性质 B: 保证对于所有  $i \neq j$ , 若  $\max(|r_i - r_j|, |c_i - c_j|) \leq 2$ , 则  $(r_i, c_i)$  与  $(r_j, c_j)$  有邻边。

特殊性质 C: 保证对于所有  $i \neq j$ ,  $\max(|r_i - r_j|, |c_i - c_j|) \neq 2$ 。

特殊性质 D: 所有特殊格子在同一行。



# 卡利贝尔 (caliber)

## 题目背景

爸爸，这太残酷了。

## 题目描述

有一颗星球上长满了人机。

这是一个长度为  $l$  的环状星球 ( $l - 1$  与  $0$  相邻)，有  $n$  个人机住在上面，第  $i$  个人机有一个只属于自己的充电站和工作点（保证所有充电站与工作点互不相同），位置分别在  $a_i, b_i$ 。

作为人机的领导，你可以控制人机的每一步走的方向以及距离。我们定义一次操作为选择一个人机使向前移动移动  $x$  的距离 ( $x \in R$ )，你可以进行这个操作无数次，直到所有人机都到达自己对应的工作点。

我们定义危险程度为在所有过程中，任意两个人机在某一时刻相遇的数量（如果一个人机在  $5$ ，另一个人机从  $3$  走到  $6$  依旧看作相遇）。

人机到达工作点后不会立即进入工作点进行工作，而是留守在门外（也就是说依旧会和其他人机产生危险程度）。

作为人机国的人机之王（赖神），需要询问你最小的危险程度是多少。

## 输入格式

从文件 `caliber.in` 中读入数据。

第一行两个整数  $n, l$ 。

接下来  $n$  行每行两个整数表示  $a_i, b_i$ 。

## 输出格式

输出到文件 `caliber.out` 中。

一行一个整数表示答案。

## 输入输出样例

### 样例输入 #1

```
1 | 3 100
2 | 10 50
3 | 30 20
4 | 60 40
```

### 样例输入 #1

```
1 | 0
```

### 样例输入 #2

1	4	100
2	30	70
3	10	12
4	60	75
5	90	50

样例输出 #2

1	1
---	---

样例组 #3

见下发文件中的 `caliber/caliber3.in`, `caliber/caliber3.out`。

该组样例满足 Subtask #3 的限制。

样例组 #4

见下发文件中的 `caliber/caliber4.in`, `caliber/caliber4.out`。

该组样例满足 Subtask #5 的限制。

样例解释 #1

第三个人机先从 60 往左边走走到 40。

第一个人机再从 10 往左边走走到 50。

第二个人机再从 30 往左边走走到 20。

不会产生任何交叉，所以答案为 0。

数据范围与约定

对于 100% 的数据， $1 \leq n \leq 10^6$ ,  $1 \leq l \leq 10^9$ ,  $0 \leq a_i, b_i < l$ ,  $a, b$  中元素互不相同， $a$  和  $b$  中不存在相同元素。

子任务编号	$n$	特殊性质	分值
Subtask #1	$\leq 10$	无	10
Subtask #2	$\leq 10^5$	保证 $a_i < a_{i+1} < b_{i+1} < b_i$	10
Subtask #3	$\leq 5000$	无	25
Subtask #4	$\leq 10^5$	保证 $a_i < a_{i+1} < b_i < b_{i+1}$	10
Subtask #5	$\leq 10^5$	无	25
Subtask #6	$\leq 10^6$	无	20