

# NOIP2024 模拟赛

GDFZ

时间：2024 年 11 月 11 日

题目名称	大战波特	决胜数论	未来都市	数据结构
题目类型	传统题	传统题	传统题	传统题
目录	bot	math	metro	ds
可执行文件名	bot	math	metro	ds
输入文件名	bot.in	math.in	metro.in	ds.in
输出文件名	bot.out	math.out	metro.out	ds.out
每个测试点时限	1 秒	1 秒	2 秒	1 秒
内存限制	512 MB	512 MB	1024 MB	512 MB
子任务数目	4	5	6	5
测试点是否等分	否	否	否	否

提交源程序文件名

对于 C++ 语言	bot.cpp	math.cpp	metro.cpp	ds.cpp
-----------	---------	----------	-----------	--------

编译选项

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

注意事项（请仔细阅读）

- 1. 测试机器：CPU(AMD Ryzen 5 3600 6-Core Processor \*12), RAM 8.0G。
- 2. 系统环境：NOI Linux 2.0(基于 Ubuntu 20.04.1)。
- 3. 文件名（程序名和输入输出文件名）必须使用英文小写。
- 4. C/C++ 中函数 `main()` 返回类型必须是 `int`，程序正常结束返回值必须是 0。
- 5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
- 6. 选手提交的程序源文件必须不大于 100KB。
- 7. 程序可使用的栈空间内存限制与题目的内存限制一致。

# 大战波特 (bot)

## 【题目描述】

你在大战波特！你要和波特比赛做题！

龟有一个  $n$  行  $m$  列的网格图，每个格子上有一个字符：**D**、**R** 或 **X**。

你要从  $(1,1)$  走到  $(n,m)$ ，每次可以从  $(x,y)$  走到  $(x+1,y)$  或  $(x,y+1)$ 。然而，如果格子上的字符是 **D** 那么你只能走到  $(x+1,y)$ ，如果是 **R** 则只能走到  $(x,y+1)$ ，如果是 **X** 则你可以自由选择。龟想考考你知不知道总共有多少不同的路径。

但是贼达不留锐锡觉得这太简单了，于是他偷偷把一些格子的字符擦掉了，只告诉你恰好  $k$  个格子的字符，他要求你求出所有  $3^{nm-k}$  种可能的网格图的答案的和 mod 998244353 的结果！

波特很菜，他只会爆搜，但它的爆搜很牛，能用恰好一秒跑出答案！为了打败他，你需要写一个能在一秒以内得出答案的程序！

## 【输入格式】

从文件 *bot.in* 中读入数据。

第一行三个整数  $n,m,k$ 。

接下来  $k$  行，每行包括  $x,y,c$ ，表示  $(x,y)$  上的字符是  $c$ 。

## 【输出格式】

输出到文件 *bot.out* 中。

一行一个整数，表示答案。

## 【样例 1 输入】

```
1 2 2 3
2 1 1 X
3 2 1 R
4 2 2 R
```

【样例 1 输出】

15

【样例 1 解释】

- 只有  $(1, 2)$  的字符不确定。
- 若  $(1, 2)$  的字符是 **R**，波特有 1 种方案可以到达  $(n, m)$ 。
  - 若  $(1, 2)$  的字符是 **D**，波特有 2 种方案可以到达  $(n, m)$ 。
  - 若  $(1, 2)$  的字符是 **X**，波特有 2 种方案可以到达  $(n, m)$ 。
- 方案数的和为 5。

【样例 2 输入】

13 3 5  
22 3 D  
31 3 D  
42 1 D  
51 2 X  
63 1 R

【样例 2 输出】

1150

【样例 3】

见选手目录下的 *bot/bot3.in* 与 *bot/bot3.ans*。

【样例 4】

见选手目录下的 *bot/bot4.in* 与 *bot/bot4.ans*。

【样例 5】

见选手目录下的 *bot/bot5.in* 与 *bot/bot5.ans*。

【数据范围】

对于所有数据，满足  $2 \leq n, m \leq 5000$ ,  $0 \leq k \leq \min(nm, 2 \times 10^5)$ ,  $1 \leq x_i \leq n$ ,  $1 \leq y_i \leq m$ , 对于任意  $i \neq j$  满足  $(x_i, y_i) \neq (x_j, y_j)$ ,  $c \in \{\mathbf{R}, \mathbf{D}, \mathbf{X}\}$ 。

子任务编号	$n \leq$	$m \leq$	$k \leq$	分值
1	3	3	9	27
2	5000	5000	0	14
3	200	200	200	23
4	5000	5000	$2 \times 10^5$	36

## 决胜数论 (math)

### 【题目背景】

众所周知，一道数学题应该有一个形式化的题面！

### 【题目描述】

对于正整数  $d$ ，定义  $d \mid n$  当且仅当  $\exists q \in \mathbb{N}^+$  满足  $d \cdot q = n$ ；

定义  $\gcd(n, m) := \max(\{d \mid d \in \mathbb{N}^+ \wedge d \mid n \wedge d \mid m\})$ ；

定义艾佛森括号  $[cond]$ ，其值为 1 当且仅当逻辑表达式  $cond$  为真，否则为 0；

定义  $n \perp m$  当且仅当  $\gcd(n, m) = 1$ ；

$$\text{定义 } F(x, n) = \begin{pmatrix} [x+1 \perp x+1] & [x+1 \perp x+2] & \cdots & [x+1 \perp x+n] \\ [x+2 \perp x+1] & [x+2 \perp x+2] & \cdots & [x+2 \perp x+n] \\ \vdots & \vdots & \ddots & \vdots \\ [x+n \perp x+1] & [x+n \perp x+2] & \cdots & [x+n \perp x+n] \end{pmatrix}。$$

(其中  $x \in \mathbb{N}$ ,  $n \in \mathbb{N}^+$ 。)

给定  $n$  和一个  $n \times n$  的矩阵  $M$  (下标从 0 开始), 求是否存在一个  $x$  使得  $F(x, n) = M$ 。  
若存在, 请输出 Yes; 否则输出 No。你需要对于  $T$  组数据求解。

### 【输入格式】

从文件 **math.in** 中读入数据。

第一行一个整数  $T$ ，表示数据组数。

对于每组数据：

第一行一个整数  $n$ ；

接下来  $n$  行，每行一个长度为  $n$  的 **01** 串，表示矩阵。

### 【输出格式】

输出到文件 **math.out** 中。

$T$  行，每行一个字符串，表示答案。

【样例 1 输入】

```
1 4
2 5
3 01010
4 10110
5 01010
6 11101
7 00010
8 2
9 01
10 00
11 6
12 011110
13 101000
14 110111
15 101010
16 101101
17 001010
18 2
19 00
20 00
```

【样例 1 输出】

```
1 Yes
2 No
3 Yes
4 No
```

【样例 1 解释】

对于第一组数据，存在  $x = 1$  使得  $F(x, n) = M$ 。

【样例 2】

见选手目录下的 `math/math2.in` 与 `math/math2.ans`。

【样例 3】

见选手目录下的 `math/math3.in` 与 `math/math3.ans`。

【样例 4】

见选手目录下的 `math/math4.in` 与 `math/math4.ans`。

【样例 5】

见选手目录下的 `math/math5.in` 与 `math/math5.ans`。

【样例 6】

见选手目录下的 `math/math6.in` 与 `math/math6.ans`。

【数据范围】

对于所有数据，满足  $1 \leq T \leq 1000$ ， $2 \leq n, \sum n \leq 5000$ ， $M_{i,j} \in \{0, 1\}$ 。

子任务编号	$n \leq$	特殊性质	分值
1	2	无	14
2	5000	$M_{0,0} = 1$	9
3	6	无	24
4	40	无	24
5	5000	无	29

## 未来都市 (metro)

### 【题目背景】

在未来社会，人类已经建成了超大型都市，在一个城市里就能领略不同国家的异域风情和生活格调。

### 【题目描述】

龟和贼达不留锐锡准备去名为“世界树”的未来都市旅游，但是他们还没设计好旅游路线。

在这个都市中，有  $n$  个国家，这  $n$  个国家由  $n - 1$  条道路连通。

龟和贼达不留锐锡已经做足了攻略，调查出了每个国家的**繁华度**： $a_i$ ，并准备从  $S$  号国家开始旅游。

龟特别讨厌繁华程度高的国家，他认为这些国家太喧闹；而贼达不留锐锡特别讨厌繁华程度低的国家，他认为这些国家太低能。

最终，他们决定，记  $P(u, v)$  为国家  $u$  到国家  $v$  的路径，定义一段行程的**恼火值**  
 $w(u, v) = \max_{x \in P(u, v)} a_x \times \min_{y \in P(u, v)} a_y$ 。

记一种**旅游方案**为  $t_0, t_1, \dots, t_k$ ，其中  $t_0 = S$ ， $k$  为任意非负整数，定义**旅游方案的恼火值**  $W(t_0, \dots, t_k) = \sum_{i=1}^k w(t_{i-1}, t_i)$ 。

龟和贼达不留锐锡想知道，对于每一个旅游的终点  $t_k = 1, \dots, n$ ，在所有可能的旅游方案中最小的**恼火值**是多少。

他们准备去  $T$  个不同的城市旅游，你需要解决全部  $T$  组数据。

### 【输入格式】

从文件 `metro.in` 中读入数据。

第一行一个整数  $T$ ，表示数据组数。

对于每组数据：

第一行两个整数  $n, S$  表示国家数和起点编号。

第二行  $n$  个整数，第  $i$  个整数表示  $a_i$ 。

接下来  $n - 1$  行，每行两个整数  $(u, v)$ ，表示国家  $u$  和国家  $v$  之间有一条道路。

### 【输出格式】

输出到文件 `metro.out` 中。

$T$  行，每行  $n$  个整数，第  $i$  个整数表示  $t_k = i$  的答案。



【样例 1 输入】

```
1 4
2 5 3
3 1 2 3 4 5
4 1 2
5 2 3
6 3 4
7 4 5
8 6 4
9 1 1 4 5 1 4
10 1 2
11 1 3
12 2 4
13 2 5
14 2 6
15 5 1
16 512 472 455 365 357
17 1 2
18 1 3
19 1 4
20 2 5
21 2 2
22 19890604 19260817
23 1 2
```

【样例 1 输出】

```
1 3 5 0 7 8
2 5 5 5 0 5 5
3 0 241664 232960 186880 182784
4 383109283663468 0
```

【样例 2】

见选手目录下的 *metro/metro2.in* 与 *metro/metro2.ans*。

【样例 3】

见选手目录下的 `metro/metro3.in` 与 `metro/metro3.ans`。

【样例 4】

见选手目录下的 `metro/metro4.in` 与 `metro/metro4.ans`。

【样例 5】

见选手目录下的 `metro/metro5.in` 与 `metro/metro5.ans`。

【样例 6】

见选手目录下的 `metro/metro6.in` 与 `metro/metro6.ans`。

【数据范围】

对于所有数据，满足  $1 \leq T \leq 10^5$ ， $2 \leq n, \sum n \leq 5 \times 10^5$ ， $1 \leq S \leq n$ ， $1 \leq a_i \leq 10^9$ ，输入的边构成一棵树。

子任务编号	$n \leq$	$\sum n \leq$	特殊性质	分值
1	10	$5 \times 10^5$	无	13
2	$5 \times 10^5$	$5 \times 10^5$	AB	8
3	$5 \times 10^5$	$5 \times 10^5$	A	22
4	$5 \times 10^5$	$5 \times 10^5$	B	22
5	$10^5$	$10^5$	无	14
6	$5 \times 10^5$	$5 \times 10^5$	无	21

- 特殊性质 A：树是一条链。
- 特殊性质 B： $\forall i \in [1, n], a_i = i$ 。

# 数据结构 (ds)

## 【题目背景】

*In computer science, a data structure is a data organization and storage format that is usually chosen for efficient access to data. More precisely, a data structure is a collection of data values, the relationships among them, and the functions or operations that can be applied to the data, i.e., it is an algebraic structure about data.*

## 【题目描述】

给定长度为  $n$  的序列  $a$ ，你需要维护  $q$  次操作：

- 给定  $v$ ，将所有  $a_i$  修改为  $\min(a_i, v)$ ;
- 将所有  $a_i$  修改为  $a_i + i$ ;
- 给定  $l, r$ ，查询  $\sum_{i=l}^r a_i$ 。

## 【输入格式】

从文件 `ds.in` 中读入数据。  
第一行两个正整数  $n, q$  表示序列的长度与操作的个数。  
下面一行  $n$  个整数  $a_i$ ，表示初始序列  $a$ 。  
下面  $q$  行，每行第一个正整数  $o$  表示第  $i$  次操作的类型。  
若  $o = 1$ ，则下面紧跟一个整数  $v$ ，表示进行一次修改一。  
若  $o = 2$ ，则表示进行一次修改二。  
若  $o = 3$ ，则下面紧跟两个正整数  $l, r$ ，表示进行一次询问三。

## 【输出格式】

输出到文件 `ds.out` 中。  
若干行，每行一个整数表示查询的答案。

## 【样例 1 输入】

```
1 15 15
2 6 14 14 6 3 6 4 13 10 3 12 5 11 9 6
3 1 9
4 1 2
```

```
5 2
6 2
7 2
8 1 11
9 3 4 6
10 2
11 1 6
12 2
13 1 9
14 1 11
15 1 11
16 3 4 4
17 3 2 13
```

【样例 1 输出】

```
1 33
2 9
3 107
```

【样例 2】

见选手目录下的 *ds/ds2.in* 与 *ds/ds2.ans*。

【样例 3】

见选手目录下的 *ds/ds3.in* 与 *ds/ds3.ans*。

【样例 4】

见选手目录下的 *ds/ds4.in* 与 *ds/ds4.ans*。

【样例 5】

见选手目录下的 *ds/ds5.in* 与 *ds/ds5.ans*。

【样例 6】

见选手目录下的 *ds/ds6.in* 与 *ds/ds6.ans*。

【数据范围】

对于所有数据，满足  $1 \leq n, q \leq 2 \times 10^5$ ,  $0 \leq a_i, v \leq 10^{12}$ ,  $1 \leq l \leq r \leq n$ ,  $o \in \{1, 2, 3\}$ 。

子任务编号	$n \leq$	$q \leq$	特殊性质	分值
1	5000	5000	无	7
2	$2 \times 10^5$	$2 \times 10^5$	A	16
3	$2 \times 10^5$	$2 \times 10^5$	B	16
4	$2 \times 10^5$	$2 \times 10^5$	C	16
5	$2 \times 10^5$	$2 \times 10^5$	无	45

- 特殊性质 A:  $a_i, v$  在  $[0, 10^{12}]$  中随机生成,  $o$  在  $[1, 3]$  中随机生成,  $[l, r]$  在所有可行区间中随机生成。
- 特殊性质 B:  $o \in \{1, 3\}$ 。
- 特殊性质 C:  $\forall i \in [1, n - 1], a_i \leq a_{i+1}$ 。