RYOI2018R2

Cmd2001 2018.05.22

|  |  |  |  |
| --- | --- | --- | --- |
| 题目名称 | 补番计划 | 推gal自动机 | 前前前任 |
| 程序文件 | bangeki | galauto | previous |
| 输入文件 | bangeki.in | galauto.in | Previous.in |
| 输入文件 | bangeki.out | galauto.out | previous.out |
| 时间限制 | 3s | 1s | 1s |
| 空间限制 | 40mb | 256mb | 128mb |
| 编译选项 | g++ -std=c++11 –O2 | g++ -std=c++11 –O2 | g++ -std=c++11 –O2 |

评测环境：

Deepin 15.5 unstable (Debian 9) amd64 with GCC 6.4.0 and Lemon 1.2.0 Beta

或

EZOJ(Ubuntu 17.10 amd64)

考生须知：

1. 严禁在考试期间交流题目，如题目有问题请咨询出题人。
2. 即使你AK了，也不要剥夺别人独立AK的权利。
3. 即使你爆零了，也不要影响别人AK。

补番计划[bangeki]

[题目描述]

小X，作为一只死宅，又攒了好多新番要补。

然而由于他的大多数时间都被用来学OI(推galgame)了，他补番的方案要精打细算。

于是他把所有要补的番排成一个长度为n序列，利用这个序列优化他的方案。

具体说，序列中的每部番都有一个特征值ai。

而由于他每天的心情不同，想看番的期望值k也不同。

每次，他会选择一个区间[l,r]，然后把这个区间内特征值ai和他的期望值k相同的番都补一集。

他想知道这次操作要补多少集番，来预估自己晚上有没有时间睡觉。

当然，由于某些特殊的影响，小X会放弃某个番剧i，并将其替换，于是他会把ai的值修改为x。

对于小X的每次询问操作，你都需要输出他补多少集番。

[输入描述]

第一行两个整数n，m，表示序列长度和操作次数。

第二行n个整数a1-an，表示初始序列。

接下来m行，每行一个整数o，表示操作类型。

如果o = 1，表示这是一个询问操作，接下来三个整数l，r，k，表示小X补番的区间和期望的特征值k。

如果o = 2，表示这是一个修改操作，接下来两个整数p，x，表示小X将序列第p项替换为特征值为x的番剧。

强制在线，l，r和p都要加上上一次的答案lastans并对n取模并+1，如果取模后l>r，请自行交换l，r。

[输出描述]

对于每个询问操作，输出一个整数ansi，表示小X补番的数量。

[样例输入]

[样例1]

5 5

3 3 3 8 3

2 2 8

1 1 3 3

1 1 2 5

1 1 2 3

1 2 2 5

[样例2]

10 10

2 2 6 2 3 3 2 3 2 2

2 4 2

1 2 6 2

1 7 9 2

2 8 3

1 6 7 3

1 6 8 6

1 7 9 3

2 7 3

1 1 7 2

1 4 8 3

[样例输出]

[样例1]

1

0

1

0

[样例2]

3

2

0

0

1

3

2

[数据范围及提示]

对于20%的数据，n,m<=5'000，

对于40%的数据，n,m<=50'000，

对于70%的数据，n,m<=500'000，

对于100%的数据，n,m<=7'00'000。

所有特征值ai均不超过MAX\_INT且非负。

时间限制3s，空间限制40mb。

推gal自动机[galauto]

[题目描述]

小X是一个沉迷galgame的死宅。

有一天他在攻略一款游戏时，发现这款游戏有一个可怕的无限循环。

如果要逃出这个循环，便需要回答n个问题。

这些问题的选项数量十分特殊。每个问题的选项数量与问题本身无关，而是和这个问题是第几个被回答的有关。

形式化的说法，如果当前还有i个问题需要被回答，那么游戏将为这个问题提供ci个选项。

小X发现这个游戏还有一些神奇的特性。如果当前还有i个问题需要回答，如果回答正确，则继续回答下一个问题；

而如果回答错误，则游戏会从区间[i,n]随机生成一个需要回答问题的数量size，然后从n个问题中随机选择size个问题并打乱答案，让小X继续回答。

由于小X快进了不少剧情，他当然是不知道这些问题的正确答案的。

于是他写了一个程序来回答问题，也就是每次无脑选择第一个选项！

他想知道从开始(剩余n个问题)到剩余q个问题的状态，这个程序期望进行多少次选择。当然这个数字不一定是个整数，所以请告诉小X这个数字在mod 1'000'000'009意义下的值。

由于小X的程序存在未知bug，有时候会造成游戏的ci值修改。而你，需要回答修改后询问的值。

[输入格式]

第一行两个整数n，m，表示总问题数和小X的操作次数。

第二行n个整数c1-cn，表示初始状态的选项数目。

接下来m行，每行一个整数o，表示操作类型。

如果o = 1，表示这是一个询问操作，接下来一个整数q，表示询问从开始(剩余n个问题)到剩余q个问题的状态，这个程序期望进行多少次选择。

如果o = 2，接下来两个整数p，x，表示小X的程序出现Bug，将cp修改为x。

[输出格式]

对于每个询问操作，输出一行一个整数ansi，表示询问的答案。

[样例输入]

[样例1]

3 3

1 4 4

1 2

1 1

1 3

[样例2]

3 5

5 3 3

2 1 1

1 3

1 1

2 3 4

1 2

[样例3]

10 10

1 5 4 5 3 3 3 1 1 4

1 5

2 5 4

2 3 2

1 2

1 1

2 10 4

1 7

1 8

1 6

2 6 1

[样例输出]

[样例1]

4

14

0

[样例2]

0

9

4

[样例3]

100000033

296429348

740876293

6

5

500000018

[数据范围及提示]

|  |  |  |  |
| --- | --- | --- | --- |
| 测试点编号 | n<= | m<= | 是否包含修改 |
| 1 | 3 | 300000 | N |
| 2 | 3 | 300000 | Y |
| 3 | 10 | 10 | N |
| 4 | 100 | 150000 | N |
| 5 | 100 | 150000 | N |
| 6 | 100 | 150000 | N |
| 7 | 100 | 10 | Y |
| 8 | 100 | 10 | Y |
| 9 | 100 | 10 | Y |
| 10 | 5000 | 150000 | N |
| 11 | 5000 | 150000 | N |
| 12 | 300 | 300 | Y |
| 13 | 300 | 300 | Y |
| 14 | 5000 | 5000 | Y |
| 15 | 5000 | 5000 | Y |
| 16 | 50000 | 50000 | N |
| 17 | 50000 | 50000 | N |
| 18 | 50000 | 50000 | Y |
| 19 | 50000 | 50000 | Y |
| 20 | 50000 | 50000 | Y |
| 21 | 100000 | 150000 | N |
| 22 | 100000 | 150000 | N |
| 23 | 100000 | 150000 | Y |
| 24 | 100000 | 150000 | Y |
| 25 | 100000 | 150000 | Y |

难度与测试点顺序不一定相关。

所有选项数量ci均不超过模数。

时间限制1s，空间限制256mb。

前前前任[previous]

[题目描述]

萌豚小X又双叒叕换老婆啦！

虽然小X说很多妹子都是他的老婆，然而这些妹子有一个共同的特点:她们都不喜欢小X！(小X:扎心了老铁)

于是她们决定，联合起来制裁一下小X这个死废宅。

小X的计算机一共有n个计算节点，为了保证安全，他会把每个妹子存储(囚禁？)在两个计算节点里。

而那个妹子一旦发起行动，就能瘫痪掉这两个计算节点。

小X的“老婆”们想知道，她们最少需要多少人，就能使得小X的计算机完全瘫痪。

保证每个计算节点里面至少存在一个妹子(即一定存在可行解)。

[输入格式]

第一行两个整数n，m，表示小X的计算机的节点总数和小X的“老婆”的数量。

接下来m行每行两个整数a，b，第i行表示第i-1个妹子被存储在a，b两个节点。

[输出格式]

一行一个整数ans，表示至少需要ans人就能使小X的计算机完全瘫痪。

[样例输入]

5 4

1 2

1 3

1 4

1 5

[样例输出]

4

[数据范围及提示]

对于前5%的数据，n<=5。

对于另外15%的数据，n<=10。

对于另外20%的数据，n<=16。

对于另外30%的数据，小X只会把妹子存放在奇偶性不同的节点中。

对于100%的数据，n<=500,m<=1000。

题目保证每个妹子的两个节点一定不同。