

NOIP2018 模拟题

Hermera

2018 年 10 月 3 日

评测时打开 O2 优化。

1 Circle

circle.cpp/in/out

Time limit: 1s

Memory limit: 512MB

1.1 Description

小 w 的男朋友送给小 w 一个 n 个点 m 条边的图，并且刁难小 w 要她找出点数最少的正环。
小 w 不会做，于是向你求助。

1.2 Input Format

第一行两个整数 n, m 。

接下来 m 行，每行四个数 u, v, a, b ，表示从 u 走到 v 的代价为 a ，从 v 走到 u 的代价为 b （算作两条不同的边）。注意 a, b 可能为负。

1.3 Output Format

当图中包含正环时，输出点数最少的正环（简单环）的点数。
否则输出 0。

1.4 Sample

1.4.1 Input

```
3 3
1 2 2 -1
2 3 10 -10
3 1 10 -10
```

1.4.2 Output

```
2
```

1.5 Constraints

对于前 20% 的数据， $n \leq 7, m \leq 10$ 。

对于前 60% 的数据， $n \leq 150, m \leq 2000$ 。

对于 100% 的数据， $1 \leq n \leq 300, 0 \leq m \leq \frac{n(n-1)}{2}, |a|, |b| \leq 10^4$ 。

数据保证不存在重边和自环。

2 Max

max.cpp/in/out

Time limit: 4s

Memory limit: 512MB

2.1 Description

小 h 的男朋友送给小 h 一个长度为 n 的序列，并且刁难小 h 要她找出其中 m 个区间的最大值。

小 h 不会做，于是向你求助。

2.2 Input Format

为了避免输入数据过大，本题使用如下方法进行输入：

第一行两个数 n, m 。其中保证 $n = 2^k, k \in \mathbb{N}$ 。

第二行三个数，分别表示 $gen, p1, p2$ 。

接下来生成 n 个数，表示长度为 n 的序列。

接下来生成 $2m$ 个数，每次两个，分别表示 m 个区间的左右端点。若第一个数大于第二个数，则交换这两个数。

生成一个数的方法为调用 `number()` 函数，其返回值为当前生成的数：

```
int gen, p1, p2;
int number() {
    gen = (1LL * gen * p1) ^ p2;
    return (gen & (n - 1)) + 1;
}
```

下发文件中将包含示例程序以便理解输入格式。

2.3 Output Format

为了避免输出数据过大，本题使用如下方法进行输出：

设 ans_i 为第 i 个区间的最大值，你只需要输出一个数：

$$\sum_{i=1}^n ans_i \cdot p1^{n-i+1} \mod p2$$

下发文件中将包含示例程序以便理解输出格式。

2.4 Sample 1

2.4.1 Input

```
4 5
32 17 19
```

2.4.2 Output

```
17
```

2.5 Sample 2

2.5.1 Input

8388608 8000000

95 1071 1989

2.5.2 Output

153

2.6 Constraints

本题共十个数据， n, m 的范围如下表：

数据点	n	m
1	8	8
2	64	50
3	128	100
4	1024	1000
5	4096	4800
6	65536	65000
7	262144	260000
8	1048576	1000000
9	8388608	7000000
10	8388608	8000000

3 Seq

seq.cpp/in/out

Time limit: 1s

Memory limit: 512MB

3.1 Description

小 y 的男朋友送给小 y 一个数列 $\{a_i\}$ ，并且刁难小 y 要她维护这个序列。

具体而言，小 y 的男朋友要求小 y 完成两个操作：

1. 修改数列中的一个数
2. 设 p_i 表示 $\max_{j=1}^i a_j$ ，求出 $\sum_{i=1}^n p_i$ 。

小 y 不会做，于是向你求助。

3.2 Input Format

第一行一个数 n 表示数列长度。

第二行 n 个由空格隔开的数表示数列 a 。

第三行一个数 m 表示修改数。

接下来 m 行，每行两个数 $pos, value$ ，表示把 a_{pos} 改成 $value$ 。

3.3 Output Format

m 行，每行一个数，表示对于每次修改后的 $\sum_{i=1}^n p_i$ 。

3.4 Sample

3.4.1 Input

```
10
114 357 904 407 100 624 449 897 115 846
20
5 357
6 350
2 939
9 1182
7 1062
2 3300
4 6867
4 2076
3 8458
9 6575
10 5737
10 338
```

9 10446
4 7615
2 5686
4 10091
1 6466
6 15551
3 10914
7 3234

3.4.2 Output

7703
7703
8565
9051
9297
29814
54783
29814
71078
71078
71078
71078
75054
75054
77440
85605
92737
119327
123429
123429

3.5 Constraints

对于前 30% 的数据, $n, m \leq 5000$;
对于前 60% 的数据, $n, m \leq 50000$;
对于 100% 的数据, $n \leq 3 \cdot 10^5$, $a_i \leq 10^9$.