

模拟赛

cdqz

时间：2021 年 6 月 26 日

一、题目概况

题目名称	卡牌高手	文明	礼物
题目类型	传统型	传统型	传统型
目录	<code>biden</code>	<code>civilization</code>	<code>gift</code>
提交文件名	<code>biden.cpp</code>	<code>civilization.cpp</code>	<code>gift.cpp</code>
输入文件名	<code>biden.in</code>	<code>civilization.in</code>	<code>gift.in</code>
输出文件名	<code>biden.out</code>	<code>civilization.out</code>	<code>gift.out</code>
每个测试点时限	1s	3s	2s
内存限制	1024MB	512MB	512MB
子任务数目	5	5	5
编译选项	<code>-lm -O2 -std=c++11</code>		

二、注意事项

1. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
2. 函数 `main()` 的返回值类型必须是 `int`，值必须为 `0`。
3. 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格进行分隔。
4. 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。
5. 程序可使用的栈空间大小与该题内存空间限制一致。
6. 在终端下可使用命令 `ulimit -s unlimited` 将栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
7. 题目按照目录名字典序排序，不一定按照题目难度排序。
8. 选手提交的源文件应直接存放在对应的个人文件夹中，无需建立对应题目的子文件夹。
9. 评测系统配置为 Ubuntu 20.04.2 LTS, Intel® Core™ i5-6500 CPU @ 3.20GHz × 4, 编译器为 g++-10 (Ubuntu 10.2.0-5ubuntu1~20.04) 10.2.0, 评测系统为 Project LemonLime 0.2.10:120, 上述时间限制以此配置为准。

卡牌高手 (biden)

【题目描述】

在春熙路，有一家神秘的咖啡厅。客人总是喜欢点一些奇怪的饮品然后在楼下汉服街随机游走。作为老板娘的好友，你需要帮她整理原材料，以及回答客人刁钻的询问。

具体地，有 n 种原材料，放 1 份原材料 i 在饮品中会产生 w_i 单位的奇异口感。由于咖啡厅刚刚开张，所以一开始没有任何原材料。

接下来，你需要处理 m 起事件。

1. 老板娘购买了 c 份原材料 i 。
2. 老板娘消耗了 c 份原材料 i 。
3. 一位客人向你询问，只使用原材料 $[l, r]$ ，咖啡厅能调配出多少种不同的饮品。由于客人是碳基生物，调配出的饮品的奇异口感不能超过 k 单位。

一杯饮品可以看做原材料的非空多重集，它的奇异口感为它原材料的奇异口感的和（如果一种原材料被使用了多份，则奇异口感会累加多次）。两杯饮品不同，当且仅当它们原材料的多重集不同。

【输入格式】

从文件 **biden.in** 中读入数据。

第一行两个正整数 n, m ，表示原材料种类数、事件数。

第二行 n 个正整数 w_1, w_2, \dots, w_n ，分表表示放 1 份每种原材料产生的奇异口感。

接下来 m 行，每一行表示一起事件，分别为以下三种格式之一：

- 三个正整数 $1, i, c$ ，表示一次购买；
- 三个正整数 $2, i, c$ ，表示一次消耗；
- 四个正整数 $3, l, r, k$ ，表示一起询问。

【输出格式】

输出到文件 **biden.out** 中。

对于每一起询问事件，输出一行一个非负整数，表示能调配出多少种不同的饮品。由于答案可能十分巨大，你只需要输出其对 998244353 取模的值。

【样例 1 输入】

```
1 5 10
2 1 3 5 2 5
```

3	1	1	4
4	1	2	5
5	3	1	5 1
6	3	1	3 6
7	3	1	5 10
8	3	1	2 20
9	2	2	2
10	3	1	1 2
11	3	2	2 8
12	3	1	5 5000

【样例 1 输出】

1	1
2	9
3	16
4	29
5	2
6	2
7	19

【样例 1 解释】

对于第 2 起询问：将饮品看做原材料的多重集，那么老板娘可以调配出以下饮品：

$$\{1\}, \{2\}, \{1, 1\}, \{1, 2\}, \{2, 2\}, \{1, 1, 1\}, \{1, 1, 2\}, \{1, 1, 1, 1\}, \{1, 1, 1, 2\}.$$

【样例 2】

见选手目录下的 *biden/biden2.in* 与 *biden/biden2.ans*。

【样例 3】

见选手目录下的 *biden/biden3.in* 与 *biden/biden3.ans*。

【测试点约束】

对于所有测试点，满足 $1 \leq n, m, w_i, c, k \leq 5000$, $1 \leq i \leq n$, $1 \leq l \leq r \leq n$ 。保证 c 份原材料 i 被消耗前至少有 c 份原材料 i 。保证在任意时刻仓库中所有原材料的奇异口感之和不

超过 5000。

每个子任务的具体限制见下表：

子任务编号	特殊限制	分值
1	$n = 14, m = 514, w_i = 1$	10
2	$n, m, w_i \leq 1000$ ，且所有购买和消耗在询问之前	20
3	$n, w_i \leq 1000, m \leq 300$	30
4	$n, m, w_i \leq 1000$	30
5	无	10

文明 (civilization)

【题外话】

出题人想写一个和文明相关的题面，但是他并没有玩过，所以只能乱糊。

【题目描述】

你正在打文明 6。你的国家采用激进型发展策略：最初，你国家的科技水平等级为最低等级 0。在第 i 年的上半年，你国家的科技水平等级会提升一级；而在下半年，你国家的科技水平会由于前半年的过度开发而跌落，任意变为一个不超过当前科技水平等级的某一等级。也就是说，如果用 p_i 代表你的国家 i 年后的科技水平等级，则 p_{i+1} 为 $[0, p_i + 1]$ 间的任意一个自然数。

B 国是与你相邻的国家，它采用稳定性发展策略：每一年科技水平等级都会稳步增加 1。但这有一定的条件：它需要 t 年来成为发达国家，然后才会开始实行该发展策略。也就是说，如果用 q_i 代表 B 国 i 年后的科技水平等级，则 $q_t = 0$ ，对于 $i > t$ 有 $q_i = q_{i-1} + 1$ 。

游戏刚开始以及每一年结束后，如果 B 国已经成为了发达国家且科技水平等级与你的国家的科技水平等级相同，那么你可以选择和 B 国开战。令 x 为你的国家当前的科技水平等级，开战后，若 $x < k$ ，游戏立即结束并且你本局会获得 v_x 的分数；否则无事发生，游戏将继续进行。其中 k 是一个固定的正整数。

现在你需要对于每一个自然数 $t \in [0, n]$ ，求出在所有可能的会结束的游戏，你获得的分数总和是多少。由于这些数过于巨大，你只需要输出它们对 998244353 取模的结果即可。

形式化地，定义一个以 x 结尾的非空自然数序列 p 的权值为

$$\begin{cases} v_x, & x < k, \\ 0, & x \geq k. \end{cases}$$

对于每一个自然数 $t \in [0, n]$ ，求出满足下列条件的自然数序列 p 的权值之和：

- $p_0 = 0$,
- $p_{i+1} \leq p_i + 1$,
- $x + t = |p|$ 。

其中 $|p|$ 表示序列 p 的长度。

【输入格式】

从文件 `civilization.in` 中读入数据。

第一行两个正整数 n, k 。

接下来一行 k 个非负整数，第 i 个整数代表 v_{i-1} 。

【输出格式】

输出到文件 *civilization.out* 中。

$n + 1$ 行每行一个非负整数，第 i 行的整数代表 $t = i - 1$ 时的答案对 998244353 取模的结果。

【样例 1 输入】

```
1 1 2
2 2 1
```

【样例 1 输出】

```
1 3
2 4
```

【样例 1 解释】

当 $t = 0$ 时， $p = [0]$ 或 $p = [0, 1]$ 时可以选择与 B 国开战，总分数为 $2 + 1 = 3$ ；当 $t = 1$ 时， $p = [0, 0]$ 、 $p = [0, 0, 1]$ 或 $p = [0, 1, 1]$ 时可以选择与 B 国开战，总分数为 $2 + 1 + 1 = 4$ 。其它情况下游戏均不能结束。

【样例 2 输入】

```
1 3 4
2 1 9 2 6
```

【样例 2 输出】

```
1 18
2 49
3 149
4 475
```

【样例 3 输入】

```
1 9 9
2 9 9 8 2 4 4 3 5 3
```

【样例 3 输出】

```
1 47
2 191
3 838
4 3552
5 14611
6 58850
7 233649
8 918548
9 3586692
10 13939750
```

【样例 4】

见选手目录下的 *civilization/civilization4.in* 与 *civilization/civilization4.ans*。

【测试点约束】

对于全部数据，有 $1 \leq n \leq 10^6$ ， $1 \leq k \leq 10^5$ ， $0 \leq v_i < 998244353$ 。
每个子任务的具体限制见下表：

子任务编号	特殊限制	分值
1	$n, k \leq 5$	10
2	$n, k \leq 500$	15
3	存在且仅存在一个 i 使得 $v_i \neq 0$	15
4	$v_i = 1$	25
5	无	35

礼物 (gift)

【题目描述】

Ame 专程乘地铁二号线来送 Mivik 礼物啦！Ame 送给 Mivik 的礼物包里，有一顶粉色的帽子，有众多精美的书签 一时间机房的大伙吃饱了狗粮。这当中还有一个竞技游戏让大伙很感兴趣，游戏的过程是这样的：

机房小 C 作为法官，先创造了一个 n 个点的树作为引，点的编号为 1 到 n 之间的正整数。小 C 先把引告诉机房小 W 和机房小 Y。小 W 会从引中选出一个点 A ，接着把和 A 点相连的边都抹去，之后打乱点的编号；小 Y 会从引中选出一个与 A 不同的点 B ，接着把和 B 点相连的边都抹去，之后打乱点的编号。然后小 W 和小 Y 同时将他们剩下的边以边集的形式（即如果某一个点不存在和它相连的点，那么它不会被表达）告诉 Mivik。现在小 C 要让 Mivik 还原出一棵满足条件的由 n 个点构成的树作为可能的引。

由于 Mivik 大部分时间都拿去和 Ame 谈恋爱去了，所以把这个游戏交给了你来完成。

【输入格式】

从文件 *gift.in* 中读入数据。

第一行一个正整数 n ，表示引的结点个数。

接下来两个部分的输入，分别表示小 W 和小 Y 告诉 Mivik 的边集合。

每个部分第一行一个非负整数 m ，表示边集的大小。接下来 m 行，每行两个正整数 a, b ，表示 a 点和 b 点之间存在一条边。

【输出格式】

输出到文件 *gift.out* 中。

输出 $n - 1$ 行，每行两个正整数，表示你还原的树的一条边。如果有多组解，输出任意一组即可。

【样例 1 输入】

```
1 5
2 3
3 1 4
4 2 1
5 5 4
6 1
7 2 5
```


【样例 1 输出】

```
1 2 3
2 3 4
3 2 5
4 2 1
```

【样例 1 解释】

如果这棵树就是引，那么 A 点编号为 1 或者 5， B 点编号为 2。

【样例 2】

见选手目录下的 `gift/gift2.in` 与 `gift/gift2.ans`。

【样例 3】

见选手目录下的 `gift/gift3.in` 与 `gift/gift3.ans`。

【测试点约束】

对于全部数据，有 $2 \leq n \leq 2000$ ， $1 \leq a, b \leq n$ ， $a \neq b$ 。保证给出的边集合法且存在至少一组解。

每个子任务的具体限制见下表：

子任务编号	特殊限制	分值
1	引是一条链	5
2	引是一个菊花图	5
3	$n \leq 10$	30
4	$n \leq 300$ ，且小 W 给出的边集构成一棵树	40
5	无	20