NOIP 2

试题

比赛时长：270 分钟　　总分：400 分

题号	题目名称	英文题名	时间限制	空间限制	评测方式	输入/输出文件
A	最小化不动点	`p16a`	3 秒	256 MiB	Special Judge	`p16a.in` `p16a.out`
B	冒泡位置	`bubble`	3 秒	256 MiB	传统型	`bubble.in` `bubble.out`
C	合成终态	`merge`	5 秒	256 MiB	传统型	`merge.in` `merge.out`
D	风后的楼	`wind`	3 秒	512 MiB	传统型	`wind.in` `wind.out`

编程语言：C++14。编译选项：`-std=c++14 -O2`。

按各题指定文件名读入与输出。各题满分均为 100 分，具体计分规则见各题数据范围与约定。

A. 最小化不动点 p16a

输入文件：p16a.in；输出文件：p16a.out。时间限制：3 秒；空间限制：256 MiB。

【题目描述】

给定正整数 n，请构造一个 1 到 n 的排列 p，即每个整数 $1, 2, \ldots, n$ 都恰好出现一次。对每个 $2 \le i \le n$，排列必须满足 $\gcd(p_i, i) > 1$。这里 gcd 表示最大公约数；位置 1 不受这条条件限制。

如果 $p_i = i$，就称位置 i 是一个不动点。你需要使不动点的数量在所有满足条件的排列中最少。若存在多个最优排列，输出任意一个。

【输入格式】

从文件 p16a.in 读入数据。

第一行包含整数 t，表示测试用例数。

接下来 t 行，每行包含一个整数 n。

【输出格式】

输出到文件 p16a.out。

每个测试用例输出一行，包含 n 个整数，表示所构造的最优排列。不需要输出不动点数。

【样例 1 输入】

```
4
2
3
6
13
```

【样例 1 输出】

```
1 2
1 2 3
1 4 6 2 5 3
1 12 9 6 10 8 7 4 3 5 11 2 13
```

【样例 1 解释】

输出中的四行分别对应四个测试用例，各行都是满足条件的最优排列。例如，第三行在位置 2 放置 4、在位置 4 放置 2，两处的最大公约数都大于 1；其不动点为位置 $1, 5$。

【数据范围与约定】

对所有数据，$1 \le t \le 10000$，$2 \le n \le 100000$。同一输入文件内所有 n 之和不超过 100000。

各测试点独立计分，无点间依赖，总分为 100 分。未另行规定的条件均与上述全局范围相同。

测试点	分值	附加限制
1	15	$n \le 8$，$t \le 5$
2	20	$n \le 18$，所有测试用例的 $n \cdot 2^n$ 之和不超过 10000000
3	25	$n \le 2000$
4	40	无附加限制

时间限制：3000 ms。空间限制：256 MiB。评测语言：C++14，编译参数为 -std=c++14 -O2。

【公开大样例附件】

以下样例各自为独立输入文件，输入与输出均在下载附件中提供。

编号	对应限制	文件
1	测试点 1 的限制	输入、输出
2	测试点 2 的限制	输入、输出
3	测试点 3 的限制	输入、输出
4	测试点 4 的限制	输入、输出

B. 冒泡位置 bubble

输入文件：bubble.in；输出文件：bubble.out。时间限制：3 秒；空间限制：256 MiB。

【题目描述】

有一个长度为 n 的排列 $a_1, a_2, \ldots, a_n$，其中 1 到 n 各出现一次。位置的下标从 1 开始。

对排列进行一轮操作，指按照 $i = 1, 2, \ldots, n-1$ 的顺序，依次执行：若当前的 $a_i > a_{i+1}$，就交换这两个数。一次交换的结果会立即用于本轮后续的比较。

例如，对排列 $(4, 3, 5, 1, 2)$ 进行一轮操作后，会得到 $(3, 4, 1, 2, 5)$。

给定 q 个询问。每个询问包含两个整数 k, x，请输出：从**给定的初始排列**出发，进行恰好 k 轮操作后，数 x 所在位置的下标。

各询问相互独立，不会改变其他询问使用的初始排列。

【输入格式】

第一行一个整数 n。

第二行 n 个整数 $a_1, a_2, \ldots, a_n$，表示初始排列。

第三行一个整数 q。

接下来 q 行，每行两个整数 k, x，表示一个询问。

【输出格式】

按输入顺序输出 q 行，每行一个整数，表示对应询问的答案。

【样例 1 输入】

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

【样例 1 输出】

```
2
4
5
```

【样例 1 解释】

初始排列以及前四轮操作后的排列依次为：

轮数	排列
0	$4, 3, 5, 1, 2$
1	$3, 4, 1, 2, 5$
2	$3, 1, 2, 4, 5$
3	$1, 2, 3, 4, 5$
4	$1, 2, 3, 4, 5$

因此，进行两轮操作后，数 1 在位置 2；
进行四轮操作后，数 4 在位置 4；进行一轮操作后，数 5 在位置 5。

【样例 2 输入】

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

【样例 2 输出】

```
2
2
3
4
5
```

【样例 3 输入】

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

【样例 3 输出】

```
1
2
3
2
1
```

【样例 3 解释】

第一轮结束后，排列为 $(2,1,3)$，数 2 位于位置 1；第二轮结束后，排列为 $(1,2,3)$，数 2 才位于位置 2。每个询问都从初始排列出发。

【数据范围与计分方式】

对于所有测试数据，保证：

- $2 \le n \le 5 \times 10^5$，$1 \le q \le 5 \times 10^5$；
- $a_1, a_2, \ldots, a_n$ 是 1 到 n 的一个排列；
- 每个询问满足 $1 \le k < n$，$1 \le x \le n$。

共有四个测试点，各自独立计分，分数相加。每个测试点的附加限制如下。

测试点	分值	附加限制
1	15	$n, q \le 2000$
2	15	所有询问中，不同的 x 不超过 20 个
3	30	$n, q \le 10^5$
4	40	无

附件提供题面中的三个普通样例、一个额外的边界样例，以及对应上述四个测试点限制的四对大样例。附件样例不参与计分。

【公开大样例附件】

以下附件各自满足对应测试点的限制，不参与正式计分。

测试点	输入	输出
1	输入	输出
2	输入	输出
3	输入	输出
4	输入	输出

附加样例：输入、输出（$n = q = 2000$，询问中不同 x 的数量不超过 20）。

附加样例：输入、输出（$n = q = 10^5$）。

C. 合成终态 merge

输入文件：merge.in；输出文件：merge.out。时间限制：5 秒；空间限制：256 MiB。

【题目描述】

有 n 颗宝石从左到右排成一列。每颗宝石有颜色和等级两种属性：颜色为 A、B、C、D 中的一种，初始等级均为 1。保证初始序列中不存在连续三颗颜色相同的宝石。

你可以进行任意轮操作，也可以不操作。每轮操作的过程如下。

1. 选择并删除一颗宝石，其右侧的宝石向左补齐。如果当前只剩一颗宝石，则不能再进行操作。
2. 如果当前存在连续三颗颜色、等级均相同的宝石，则选择其中**最靠左的一组**，将这三颗宝石替换为一颗颜色相同、等级增加 1 的宝石，其右侧的宝石向左补齐。
3. 重复第 2 步，直到不存在这样的三颗宝石，本轮操作才结束。

你可以在任意一轮操作结束后停止，也可以直接保留初始序列。每次停止时，剩下的非空宝石序列称为一个终态。

求一共可以得到多少种不同的终态，答案对 998244353 取模。两个终态不同，当且仅当宝石数量不同，或存在某个位置，两者在该位置上的宝石颜色或等级不同。得到同一终态的不同操作过程只计一次。

【输入格式】

第一行一个正整数 n。

第二行一个长度为 n、仅含 A、B、C、D 的字符串 S，其中第 i 个字符表示初始第 i 颗宝石的颜色。

【输出格式】

输出一行一个整数，表示不同终态的数量对 998244353 取模的结果。

【样例 1 输入】

```
1
A
```

【样例 1 输出】

```
1
```

【样例 1 解释】

当前只有一颗宝石，唯一的终态就是初始序列。

【样例 2 输入】

```
5
AABAA
```

【样例 2 输出】

```
13
```

【样例 2 解释】

用 $A(h)$ 表示颜色为 A、等级为 h 的宝石，其他颜色同理。

保留 $B(1)$ 时，它左右各可以保留 0、1、2 颗 $A(1)$，共有 9 种终态。

不保留 $B(1)$ 时，能够得到的终态恰有以下 4 种：

$$A(2)A(1), \qquad A(2), \qquad A(1)A(1), \qquad A(1).$$

例如，直接删除中间的 $B(1)$ 后，剩下四颗 $A(1)$。按照最靠左的三颗先合并的规则，得到 $A(2)A(1)$。

【样例 3 输入】

```
5
ABBAB
```

【样例 3 输出】

```
17
```

【数据范围与评分】

对于所有数据，$1 \le n \le 10^6$，S 的长度为 n，仅由 A、B、C、D 组成，且不存在连续三个相同的字符。

各测试点独立计分，无依赖。每个测试点同时满足全局限制与下表中的限制。

测试点	分值	附加限制
1	15	$n \le 12$
2	10	对所有 $1 \le i < n$，均有 $S_i \neq S_{i+1}$
3	15	$n \le 1000$
4	25	$n \le 10^5$
5	35	无附加限制

【公开大样例附件】

以下样例各自为独立输入文件，输入与输出均在下载附件中提供。同一限制可以对应多组样例。

对应限制	文件
测试点 1	输入、输出
测试点 2	输入、输出
测试点 3	输入、输出
测试点 4	输入、输出
测试点 4	输入、输出
测试点 5	输入、输出
测试点 5	输入、输出
测试点 5	输入、输出

D. 风后的楼 wind

输入文件：wind.in；输出文件：wind.out。时间限制：3 秒；空间限制：512 MiB。

【题目描述】

一座楼由 $n+2$ 行、m 列方格组成，行从上到下编号为 $0, 1, \ldots, n+1$，列从左到右编号为 $1, 2, \ldots, m$。最初所有方格都存在。

第 0 行和第 $n+1$ 行的方格永远不会被吹走。其余 n 行会经历连续 k 天的风。每天先吹一次左风，再吹一次右风：

- 每次左风，对每个可破坏行分别作出一个 0/1 决定。决定为 1 时，吹走该行**当前最左边**的一个方格；决定为 0 时，不吹走该行的方格。
- 每次右风也对每个可破坏行分别作出一个 0/1 决定。决定为 1 时，吹走该行**当前最右边**的一个方格；决定为 0 时，不吹走该行的方格。

一次风对各行的影响**同时发生**。若某行已没有方格，则本次风不会再吹走这一行的方格，但仍须为该行作出一个 0/1 决定。右风面对的是当天左风结束后的楼。

全部 k 天共有 $2nk$ 个 0/1 决定。只要有一个决定不同，就认为是不同的风力方案。

两个仍存在的方格在共用一条边时相邻。一种风力方案是**安全的**，当且仅当在每一次左风和右风结束后，楼中所有仍存在的方格构成一个四连通块。第 0 行和第 $n+1$ 行也属于需要检查的方格。

求安全风力方案的数量，对 10^9+7 取模。$k=0$ 时没有任何风，也没有任何决定，空方案算作一种方案。

【输入格式】

第一行一个正整数 T，表示测试数据组数。

接下来 T 行，每行三个整数 n, m, k，表示一组测试数据。

【输出格式】

输出 T 行，每行一个整数，表示对应测试数据的答案。

【样例 1 输入】

```
4
1 2 1
2 2 1
2 3 1
3 4 0
```

【样例 1 输出】

```
3
7
16
1
```

【样例 1 解释】

第一组数据只有一个可破坏行，左右风的决定共有四种。只有两次都决定吹走方格的方案不安全，因此答案为 3。

第二组数据中，每个可破坏行在当天结束时有三种非空形状：保留两格、只保留左格、只保留右格。共有 $3^2=9$ 种组合；其中上行只保留左格而下行只保留右格，以及反过来的两种组合不连通，因此答案为 7。

第三组数据中，每个可破坏行无论怎样作出两次决定，第 2 列的方格都始终存在，所以全部 $2^4=16$ 种方案都安全。

【样例 2 输入】

```
3
1 2 2
2 2 2
2 3 2
```

【样例 2 输出】

```
5
17
95
```

【样例 2 解释】

第一组数据共作出四次决定。安全方案可以一次也不吹走方格，也可以只在四次风中的某一次吹走一格，共 5 种。不同时间吹走同一个方格，仍然算作不同的方案。

【数据范围与评分】

对于所有测试数据：

- $1 \leq T \leq 10$；
- $1 \leq n, m \leq 2000$，$0 \leq k \leq 2000$；
- 同一输入文件内，$\sum nm \leq 4 \times 10^6$，$\sum km \leq 4 \times 10^6$。

各测试点独立计分，没有依赖。每个测试点中的所有测试数据都满足对应的附加限制。

测试点	分值	附加限制
1	10	$nk \leq 8$
2	20	$n = 1$
3	20	$n, m \leq 18$
4	15	$m \leq 60$
5	35	无附加限制

【公开大样例附件】

以下样例的输入、输出均在下载附件中提供，各自是独立输入文件。

编号	对应限制	输入与输出
1	测试点 1	输入、输出
2	测试点 2	输入、输出
3	测试点 3	输入、输出
4	测试点 4	输入、输出
5	测试点 5	输入、输出
6	测试点 5	输入、输出
7	测试点 5	输入、输出