

四川省省队集训

第一组

时间：1926 年 8 月 17 日 00:00 ~ 00:00:01

题目名称	字符串	矩阵	冰棒
题目类型	传统型	传统型	传统型
目录	string	matrix	popsicle
可执行文件名	string	matrix	popsicle
输入文件名	string.in	matrix.in	popsicle.in
输出文件名	string.out	matrix.out	popsicle.out
每个测试点时限	2.5 秒	1.0 秒	2.0 秒
内存限制	1024 MiB	512 MiB	512 MiB
测试点数目	0	20	25
测试点是否等分	是	是	是

提交源程序文件名

对于 C++ 语言	string.cpp	matrix.cpp	popsicle.cpp
-----------	------------	------------	--------------

编译选项

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

字符串 (string)

【题目描述】

对于一个字符串 S , 定义 $root(S)$ 表示 S 的最小整周期, $R(S) = \frac{|S|}{root(S)}$ 即最小整周期出现次数。

定义 $S[l, \dots, r]$ 表示 S 字符串第 l 个字符到第 r 个字符所构成的子串。

定义 $f(S)$ 表示 $\max_{l \leq r} R(S[l, \dots, r])$, 即 S 的子串中最大的 R 。

给定一个长度为 n 的字符串 S , 你需要找一个最大的子串 $S[L, \dots, R]$, 使其具有最大的 R , 若存在多个, 输出字典序最小的。

接下来给出 m 个询问, 每次询问给定两个整数 $l, r (1 \leq l \leq r \leq n)$, 你需要输出 $f(S[l, \dots, r])$ 。

【输入格式】

从文件 **string.in** 中读入数据。

第一行输入一个字符串 S 。

第二行输入一个整数 m 。

接下来 m 行, 每行两个整数, 表示 l, r 。

【输出格式】

输出到文件 **string.out** 中。

第一行一个字符串 T , 表示 S 中具有最大的 R 的子串, 若有多个需要输出字典序最小的。

接下来 m 行, 每行一个正整数, 表示对应询问的答案。

【样例 0 输入】

```
1 ccabababc
2 5
3 2 9
4 1 7
5 2 5
6 2 6
7 3 8
```

【样例 0 输出】

```
1 ababab
2 3
3 2
4 1
5 2
6 3
```

【样例 1】

见选手目录下的 *string/string1.in* 与 *string/string1.ans*。

【样例 2】

见选手目录下的 *string/string2.in* 与 *string/string2.ans*。

【样例 3】

见选手目录下的 *string/string3.in* 与 *string/string3.ans*。

【样例 4】

见选手目录下的 *string/string4.in* 与 *string/string4.ans*。

【样例 5】

见选手目录下的 *string/string5.in* 与 *string/string5.ans*。

【样例 6】

见选手目录下的 *string/string6.in* 与 *string/string6.ans*。

【样例 7】

见选手目录下的 *string/string7.in* 与 *string/string7.ans*。

样例 1 ~ 7 分别满足每一个子任务的限制。

【子任务】

本题使用捆绑测试。

对于所有数据，保证 $1 \leq n, m \leq 10^6, 1 \leq l \leq r \leq n$ ，保证 S 只由小写英文字母组成。

子任务编号	$n \leq$	$m \leq$	特殊性质	分值
1	500	500	无	10
2	5000	5000	无	10
3	2×10^5	0	无	10
4	10^6	0	无	20
5	10^6	10^6	A	10
6	2×10^5	2×10^5	无	20
7	10^6	10^6	无	20

特殊性质 A：保证字符串的每一个字符在小写英文字母中随机，每个询问的 $[l, r]$ 分别在 $[1, n]$ 范围内随机。

矩阵 (matrix)

【题目描述】

给定 $n \times m$ 的矩阵 A ，你需要构造 $n \times m$ 的 01 矩阵 B ，满足：

- $\forall i \in [1, n], j \in [1, m]$ ，若 $B_{i,j}$ 为 1，则 $B_{i,j-1}, B_{i,j+1}, B_{i-1,j}, B_{i+1,j}$ 均为 0。
- $\forall i \in [1, n], j \in [1, m]$ 满足 $i+j$ 为奇数，若 $B_{i,j}$ 为 1，则 $B_{i-1,j-1}, B_{i+1,j+1}$ 均为 0。
- $\forall i \in [1, n], j \in [1, m]$ 满足 $i+j$ 为偶数，若 $B_{i,j}$ 为 1，则 $B_{i-1,j+1}, B_{i+1,j-1}$ 均为 0。

对于 $i \notin [1, n]$ 或 $j \notin [1, m]$ ，我们钦定 $B_{i,j} = 0$ 。

你需要最大化 $\sum B_{i,j} \times A_{i,j}$ 。

【输入格式】

从文件 `matrix.in` 中读入数据。

对于每组数据，第一行两个整数 n, m 。

接下来 n 行，每行 m 个整数，第 i 行第 j 个整数表示 $A_{i,j}$ 。

【输出格式】

输出到文件 `matrix.out` 中。

第一行一个整数，表示最大的 $\sum B_{i,j} \times A_{i,j}$ 。

接下来 n 行，每行 m 个值为 0 或 1 的整数，第 i 行第 j 个整数表示你所构造的 $B_{i,j}$ 。

如果有多组解，你可以输出任意一种。

【样例 0 输入】

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

【样例 0 输出】

```
1 19
2 0 1 0 1
3 0 0 0 0
4 1 0 1 0
```

【样例 1】

见选手目录下的 *matrix/matrix1.in* 与 *matrix/matrix1.ans*。

【样例 2】

见选手目录下的 *matrix/matrix2.in* 与 *matrix/matrix2.ans*。

【样例 3】

见选手目录下的 *matrix/matrix3.in* 与 *matrix/matrix3.ans*。

【样例 4】

见选手目录下的 *matrix/matrix4.in* 与 *matrix/matrix4.ans*。

【样例 5】

见选手目录下的 *matrix/matrix5.in* 与 *matrix/matrix5.ans*。

【样例 6】

见选手目录下的 *matrix/matrix6.in* 与 *matrix/matrix6.ans*。

【样例 7】

见选手目录下的 *matrix/matrix7.in* 与 *matrix/matrix7.ans*。

样例 1 ~ 7 分别满足测试点 1, 2, 3, 5, 7, 9, 12 的限制。

除样例 0 外，其余样例没有给出具体构造方案。

【提示】

本题下发 SPJ。

将选手目录下的 *spj.cpp* 与 *testlib.h* 置于同一文件夹下，并在终端使用 `g++ spj.cpp -o spj -std=c++14 -O2` 编译得到可运行文件。使用 `spj <input> <output> <answer>` 运行 SPJ。

下发的 SPJ 与实际评测中使用的 有所不同。

【子任务】

对于所有数据， $1 \leq n \times m \leq 1000$ ， $1 \leq A_{i,j} \leq 10^9$ 。

测试点编号	$n \times m \leq$	特殊性质
1	1000	A
2	20	无
3 ~ 4	100	无
5 ~ 6	200	无
7 ~ 8	350	无
9 ~ 11	1000	B
12 ~ 20	1000	无

特殊性质 A：保证 $n = 1$ 。

特殊性质 B：保证 $\forall i \in [1, n], j \in [1, m], A_{i,j} = 1$ 。

冰棒 (popsicle)

【题目描述】

暑假来了，小 δ 想要吃冰棒。

小 δ 知道 n 家售卖冰棒的商店，它们被依次编号为 0 到 $(n-1)$ 。这些商店的老板都十分任性——他们每天售卖的冰棒口味并不相同。小 δ 只喜欢西瓜味的冰棒，因此他早就做好了调查。小 δ 的调查结果形如 n 个有理数序列，其中第 i 个序列的长度为 l_i ，且它的第 j 项记作 $p_{i,j}$ 。为了方便起见，我们给出的值为 $p'_{i,j}$ ，它是 $p_{i,j}$ 对 998244353 取模的结果，可以说明在本题限制下这不会影响求解的过程。此外， i, j 均从 0 开始标号。

小 δ 的暑假长达 $2025!$ 天。对于商店 i ，在第 t 天，它售卖西瓜味冰棒的几率为 $p_{i,(t \bmod l_i)}$ 。小 δ 每天都会前往所有商店购买冰棒，如果其中至少一个商店售卖西瓜味冰棒，小 δ 就会开心。

现在，小 δ 想要知道：他开心的日子占整个暑假的比例的期望是多少？为了方便你的计算，他只需要你告诉他这个答案对 998244353 取模的结果就好了。

【输入格式】

从文件 `popsicle.in` 中读入数据。

本题有多组测试数据。

输入的第一行包含正整数 T ，代表数据组数。

接下来依次输入每组测试数据，对于每组测试数据：

输入的第一行包含正整数 n 。

接下来的 n 行，每行开头包含一个正整数 l_i ，接下来的 l_i 个非负整数为 $p'_{i,j}$ 。

【输出格式】

输出到文件 `popsicle.out` 中。

对于每组测试数据，输出一行一个非负整数，代表所求的答案。

【样例 1 输入】

```
1 2
2 2
3 1 499122177
4 2 499122177 1
5 2
6 2 1 0
7 3 1 0 1
```


【样例 1 输出】

```
1 124780545
2 831870295
```

【样例 1 解释】

对于第一组数据，输入中的 $p'_{i,j} = 499122177$ 代表 $p_{i,j} = \frac{1}{2}$ 。在第 $1, 3, 5, \dots$ 天，小 δ 一定能在商店 1 购买到冰棒，在第 $2, 4, 6, \dots$ 天，小 δ 有 $\frac{3}{4}$ 的几率买到冰棒。可以说明，最后的答案应为 $\frac{7}{8}$ 对 998244353 取模的结果。

对于第二组数据，可以说明，最后的答案应为 $\frac{5}{6}$ 对 998244353 取模的结果。

【样例 2】

见选手目录下的 *popsicle/popsicle2.in* 与 *popsicle/popsicle2.ans*。
这个样例满足测试点 1 的条件限制。

【样例 3】

见选手目录下的 *popsicle/popsicle3.in* 与 *popsicle/popsicle3.ans*。
这个样例满足测试点 2 的条件限制。

【样例 4】

见选手目录下的 *popsicle/popsicle4.in* 与 *popsicle/popsicle4.ans*。
这个样例满足测试点 3 ~ 4 的条件限制。

【样例 5】

见选手目录下的 *popsicle/popsicle5.in* 与 *popsicle/popsicle5.ans*。
这个样例满足测试点 5 ~ 7 的条件限制。

【样例 6】

见选手目录下的 *popsicle/popsicle6.in* 与 *popsicle/popsicle6.ans*。
这个样例满足测试点 8 ~ 15 的条件限制。

【样例 7】

见选手目录下的 *popsicle/popsicle7.in* 与 *popsicle/popsicle7.ans*。
这个样例满足测试点 21 ~ 25 的条件限制。

【子任务】

对于所有测试数据保证： $1 \leq T \leq 5$, $1 \leq n \leq 3 \times 10^3$, $1 \leq l_i \leq 100$, $0 \leq p'_{i,j} < 998244353$ 。

测试点	特殊性质
1	$\forall 1 \leq i < j \leq n, l_i = l_j$
2	$\text{lcm}(l_1, l_2, \cdots, l_n) \leq 10^3$
3, 4	$\text{lcm}(l_1, l_2, \cdots, l_n) \leq 10^6$
5 ~ 7	$\forall 1 \leq i \leq n, l_i$ 为质数
8 ~ 15	$l_i \leq 50$
16 ~ 20	$T = 1$
21 ~ 25	无

【提示】

有理数 $\frac{a}{b}$ 对 P 取模的结果被定义为在 $[0, P)$ 内的整数 c , 满足 $c \times b \equiv a \pmod{P}$ 。
可以说明, 在 $P \nmid b$ 且 P 为质数时, 存在唯一这样的整数 c 。
本题读入量较大, 请选择合适的读入方法。