2025 CCF 非专业级软件能力认证

CSP-J/S 2025 第二轮认证

入门级

时间：2025 年 11 月 1 日 08:30 ～ 12:00

题目名称	拼数	座位	异或和	多边形
题目类型	传统型	传统型	传统型	传统型
目录	`number`	`seat`	`xor`	`polygon`
可执行文件名	`number`	`seat`	`xor`	`polygon`
输入文件名	`number.in`	`seat.in`	`xor.in`	`polygon.in`
输出文件名	`number.out`	`seat.out`	`xor.out`	`polygon.out`
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点数目	25	20	20	25
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	`number.cpp`	`seat.cpp`	`xor.cpp`	`polygon.cpp`

编译选项

对于 C++ 语言	`-O2 -std=c++14 -static`

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. `main` 函数的返回值类型必须是 `int`，程序正常结束时的返回值必须是 0。
3. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末换行）。
4. 选手提交的程序源文件大小不得超过 100 KiB。
5. 提交的程序源文件的放置位置请参考各省的具体要求。
6. 程序可使用的栈空间内存限制与题目的内存限制一致。
7. 禁止在源代码中改变编译器参数（如使用 `#pragma` 命令），禁止使用系统结构相关指令（如内联汇编）或其他可能造成不公平的方法。
8. 因违反上述规定而出现的问题，申诉时一律不予受理。
9. 只提供 Linux 格式附加样例文件。
10. 全国统一评测时采用的机器配置为：Intel Core Ultra 9 285K CPU @ 3.70 GHz（关闭睿频与能效核），内存 96 GB。上述时限以此配置为准。
11. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

拼数（number）

【题目描述】

小 R 正在学习字符串处理。小 X 给了小 R 一个字符串 s，其中 s 仅包含小写英文字母及数字，且**包含至少一个 $1 \sim 9$ 中的数字**。小 X 希望小 R 使用 s 中的**任意多个**数字，按**任意顺序**拼成一个正整数。**注意：小 R 可以选择 s 中相同的数字，但每个数字只能使用一次**。例如，若 s 为 `1a01b`，则小 R 可以同时选择第 $1, 3, 4$ 个字符，分别为 $1, 0, 1$，拼成正整数 101 或 110；但小 R 不能拼成正整数 111，因为 s 仅包含两个数字 1。

小 R 想知道，在他所有能拼成的正整数中，最大的是多少。你需要帮助小 R 求出他能拼成的正整数的最大值。

【输入格式】

从文件 ***number.in*** 中读入数据。

输入的第一行包含一个字符串 s，表示小 X 给小 R 的字符串。

【输出格式】

输出到文件 ***number.out*** 中。

输出一行一个正整数，表示小 R 能拼成的正整数的最大值。

【样例 1 输入】

```
5
```

【样例 1 输出】

```
5
```

【样例 1 解释】

s 仅包含一个数字 5，因此小 R 仅能拼成正整数 5。

【样例 2 输入】

```
290es1q0
```

【样例 2 输出】

```
92100
```

【样例 2 解释】

s 包含数字 $2, 9, 0, 1, 0$。可以证明，小 R 拼成的正整数的最大值为 92100。

【样例 3】

见选手目录下的 ***number/number3.in*** 与 ***number/number3.ans***。
该样例满足测试点 $9 \sim 11$ 的约束条件。

【样例 4】

见选手目录下的 ***number/number4.in*** 与 ***number/number4.ans***。
该样例满足测试点 20 的约束条件。

【数据范围】

设 $|s|$ 为字符串 s 的长度。对于所有测试数据，保证：

- $1 \leq |s| \leq 10^6$；
- s 仅包含小写英文字母及数字，且包含至少一个 $1 \sim 9$ 中的数字。

测试点编号	$\|s\| \leq$	特殊性质
1	1	A
2	2	
3		无
4	10	A
5, 6		无
7, 8	10^2	A
$9 \sim 11$		无
12	10^3	A
13, 14		无
15	10^5	A
16, 17		B
18, 19		无
20	10^6	A
21, 22		B
$23 \sim 25$		无

特殊性质 A：s 仅包含数字。

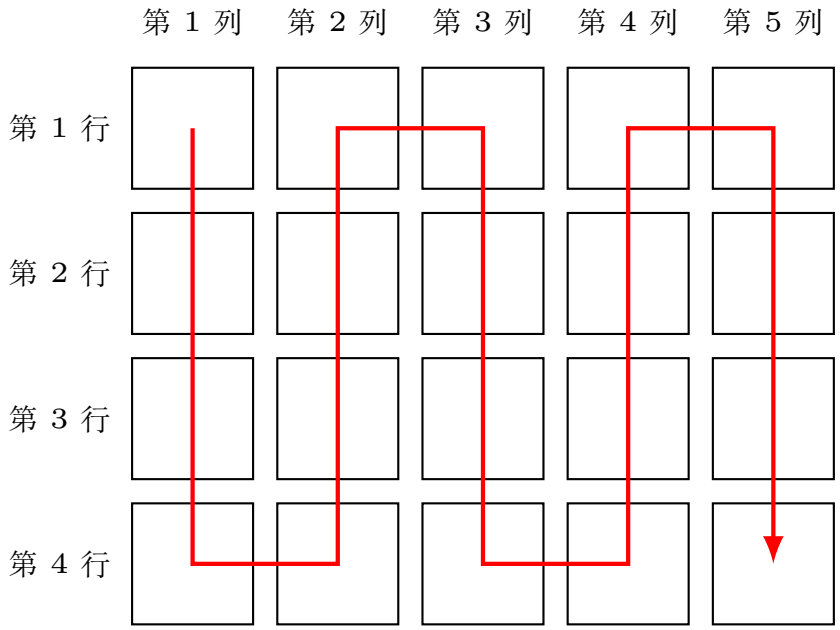
特殊性质 B：s 仅包含不超过 10^3 个数字。

座位（seat）

【题目描述】

CSP-J 2025 第二轮正在进行。小 R 所在的考场共有 $n \times m$ 名考生，其中所有考生的 CSP-J 2025 第一轮成绩**互不相同**。所有 $n \times m$ 名考生将按照 CSP-J 2025 第一轮的成绩，**由高到低蛇形分配**座位，排列成 n **行** m **列**。具体地，设小 R 所在的考场的所有考生的成绩**从高到低**分别为 $s_1 > s_2 > \cdots > s_{n \times m}$，则成绩为 s_1 的考生的座位为第 1 **列**第 1 **行**，成绩为 s_2 的考生的座位为第 1 **列**第 2 **行**，……，成绩为 s_n 的考生的座位为第 1 **列**第 n **行**，成绩为 s_{n+1} 的考生的座位为第 2 **列**第 n **行**，……，成绩为 s_{2n} 的考生的座位为第 2 **列**第 1 **行**，成绩为 s_{2n+1} 的考生的座位为第 3 **列**第 1 **行**，以此类推。

例如，若 $n = 4$，$m = 5$，则所有 $4 \times 5 = 20$ 名考生将按照 CSP-J 2025 第一轮成绩从高到低的顺序，根据下图中的箭头顺序蛇形分配座位。



给定小 R 所在的考场座位的**行数** n 与**列数** m，以及小 R 所在的考场的所有考生 CSP-J 2025 第一轮的成绩 $a_1, a_2, \ldots, a_{n \times m}$，其中 a_1 为小 R CSP-J 2025 第一轮的成绩，你需要帮助小 R 求出，他的座位为第几**列**第几**行**。

【输入格式】

从文件 ***seat.in*** 中读入数据。

输入的第一行包含两个正整数 n, m，分别表示小 R 所在的考场座位的**行数**与**列数**。

输入的第二行包含 $n \times m$ 个正整数 $a_1, a_2, \ldots, a_{n \times m}$，分别表示小 R 所在的考场的所有考生 CSP-J 2025 第一轮的成绩，其中 a_1 为小 R CSP-J 2025 第一轮的成绩。

【输出格式】

输出到文件 ***seat.out*** 中。

输出一行两个正整数 c, r，表示小 R 的座位为第 c 列第 r 行。

【样例 1 输入】

```
2 2
99 100 97 98
```

【样例 1 输出】

```
1 2
```

【样例 1 解释】

按照成绩从高到低的顺序，成绩为 100 的考生的座位为第 1 列第 1 行，成绩为 99 的考生的座位为第 1 列第 2 行，成绩为 98 的考生的座位为第 2 列第 2 行，成绩为 97 的考生的座位为第 2 列第 1 行。小 R 的成绩为 99，因此座位为第 1 列第 2 行。

【样例 2 输入】

```
2 2
98 99 100 97
```

【样例 2 输出】

```
2 2
```

【样例 2 解释】

按照成绩从高到低的顺序，成绩为 100 的考生的座位为第 1 列第 1 行，成绩为 99 的考生的座位为第 1 列第 2 行，成绩为 98 的考生的座位为第 2 列第 2 行，成绩为 97 的考生的座位为第 2 列第 1 行。小 R 的成绩为 98，因此座位为第 2 列第 2 行。

【样例 3 输入】

```
3 3
94 95 96 97 98 99 100 93 92
```

【样例 3 输出】

```
3 1
```

【数据范围】

对于所有测试数据，保证：

- $1 \le n \le 10$，$1 \le m \le 10$；
- 对于所有 $1 \le i \le n \times m$，均有 $1 \le a_i \le 100$，且 $a_1, a_2, \ldots, a_{n \times m}$ 互不相同。

测试点编号	$n \le$	$m \le$	特殊性质
1	1	1	AB
2, 3		10	无
4, 5	10	1	
6	2	2	A
7			B
8, 9			无
10		10	A
11			B
12 ∼ 14			无
15 ∼ 17	10	2	
18 ∼ 20		10	

特殊性质 A：对于所有 $1 \le i \le n \times m$，均有 $a_i = i$。

特殊性质 B：对于所有 $1 \le i \le n \times m$，均有 $a_i = n \times m - i + 1$。

异或和（xor）

【题目描述】

小 R 有一个长度为 n 的非负整数序列 $a_1, a_2, \ldots, a_n$。定义一个区间 $[l, r]$ $(1 \le l \le r \le n)$ 的权值为 $a_l, a_{l+1}, \ldots, a_r$ 的二进制按位异或和，即 $a_l \oplus a_{l+1} \oplus \cdots \oplus a_r$，其中 $\oplus$ 表示二进制按位异或。

小 X 给了小 R 一个非负整数 k。小 X 希望小 R 选择序列中尽可能多的**不相交**的区间，使得每个区间的权值均为 k。两个区间 $[l_1, r_1], [l_2, r_2]$ 相交当且仅当两个区间同时包含至少一个相同的下标，即存在 $1 \le i \le n$ 使得 $l_1 \le i \le r_1$ 且 $l_2 \le i \le r_2$。

例如，对于序列 $[2, 1, 0, 3]$，若 $k = 2$，则小 R 可以选择区间 $[1, 1]$ 和区间 $[2, 4]$，权值分别为 2 和 $1 \oplus 0 \oplus 3 = 2$；若 $k = 3$，则小 R 可以选择区间 $[1, 2]$ 和区间 $[4, 4]$，权值分别为 $1 \oplus 2 = 3$ 和 3。

你需要帮助小 R 求出他能选出的区间数量的最大值。

【输入格式】

从文件 ***xor.in*** 中读入数据。

输入的第一行包含两个非负整数 n, k，分别表示小 R 的序列长度和小 X 给小 R 的非负整数。

输入的第二行包含 n 个非负整数 $a_1, a_2, \ldots, a_n$，表示小 R 的序列。

【输出格式】

输出到文件 ***xor.out*** 中。

输出一行一个非负整数，表示小 R 能选出的区间数量的最大值。

【样例 1 输入】

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

【样例 1 输出】

```
2
```

【样例 1 解释】

小 R 可以选择区间 $[1, 1]$ 和区间 $[2, 4]$，异或和分别为 2 和 $1 \oplus 0 \oplus 3 = 2$。可以证明，小 R 能选出的区间数量的最大值为 2。

【样例 2 输入】

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

【样例 2 输出】

```
2
```

【样例 2 解释】

小 R 可以选择区间 $[1,2]$ 和区间 $[4,4]$，异或和分别为 $1 \oplus 2 = 3$ 和 3。可以证明，小 R 能选出的区间数量的最大值为 2。

【样例 3 输入】

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

【样例 3 输出】

```
1
```

【样例 3 解释】

小 R 可以选择区间 $[3,3]$，异或和为 0。可以证明，小 R 能选出的区间数量的最大值为 1。注意：小 R 不能同时选择区间 $[3,3]$ 和区间 $[1,4]$，因为这两个区间同时包含下标 3。

【样例 4】

见选手目录下的 ***xor/xor4.in*** 与 ***xor/xor4.ans***。
该样例满足测试点 4, 5 的约束条件。

【样例 5】

见选手目录下的 ***xor/xor5.in*** 与 ***xor/xor5.ans***。
该样例满足测试点 9, 10 的约束条件。

【样例 6】

见选手目录下的 ***xor/xor6.in*** 与 ***xor/xor6.ans***。
该样例满足测试点 14, 15 的约束条件。

【数据范围】

对于所有测试数据，保证：

- $1 \le n \le 5 \times 10^5$，$0 \le k < 2^{20}$；
- 对于所有 $1 \le i \le n$，均有 $0 \le a_i < 2^{20}$。

测试点编号	$n \le$	k	特殊性质
1	2	$= 0$	A
2	10	≤ 1	B
3	10^2	$= 0$	A
4, 5		≤ 1	B
6 ∼ 8		≤ 255	C
9, 10	10^3		
11, 12		$< 2^{20}$	无
13	2×10^5	≤ 1	B
14, 15		≤ 255	C
16		$< 2^{20}$	无
17	5×10^5	≤ 255	C
18 ∼ 20		$< 2^{20}$	无

特殊性质 A：对于所有 $1 \le i \le n$，均有 $a_i = 1$。
特殊性质 B：对于所有 $1 \le i \le n$，均有 $0 \le a_i \le 1$。
特殊性质 C：对于所有 $1 \le i \le n$，均有 $0 \le a_i \le 255$。

多边形（polygon）

【题目描述】

小 R 喜欢玩小木棍。小 R 有 n 根小木棍，第 i $(1 \le i \le n)$ 根小木棍的长度为 a_i。

小 X 希望小 R 从这 n 根小木棍中选出若干根小木棍，将它们按任意顺序首尾相连拼成一个多边形。小 R 并不知道小木棍能拼成多边形的条件，于是小 X 直接将条件告诉了他：对于长度分别为 $l_1, l_2, \ldots, l_m$ 的 m 根小木棍，这 m 根小木棍能拼成一个多边形当且仅当 $m \ge 3$ 且所有小木棍的长度之和**大于**所有小木棍的长度最大值的两倍，即 $\sum_{i=1}^{m} l_i > 2 \times \max_{i=1}^{m} l_i$。

由于小 R 知道了小木棍能拼成多边形的条件，小 X 提出了一个更难的问题：有多少种选择小木棍的方案，使得选出的小木棍能够拼成一个多边形？你需要帮助小 R 求出选出的小木棍能够拼成一个多边形的方案数。两种方案不同当且仅当选择的小木棍的**下标集合不同**，即存在 $1 \le i \le n$，使得其中一种方案选择了第 i 根小木棍，但另一种方案未选择。由于答案可能较大，你只需要求出答案对 $998,244,353$ 取模后的结果。

【输入格式】

从文件 ***polygon.in*** 中读入数据。

输入的第一行包含一个正整数 n，表示小 R 的小木棍的数量。

输入的第二行包含 n 个正整数 $a_1, a_2, \ldots, a_n$，表示小 R 的小木棍的长度。

【输出格式】

输出到文件 ***polygon.out*** 中。

输出一行一个非负整数，表示小 R 选出的小木棍能够拼成一个多边形的方案数对 $998,244,353$ 取模后的结果。

【样例 1 输入】

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

【样例 1 输出】

```
9
```

【样例 1 解释】

共有以下 9 种选择小木棍的方案，使得选出的小木棍能够拼成一个多边形：

1. 选择第 $2,3,4$ 根小木棍，长度之和为 $2+3+4=9$，长度最大值为 4；
2. 选择第 $2,4,5$ 根小木棍，长度之和为 $2+4+5=11$，长度最大值为 5；
3. 选择第 $3,4,5$ 根小木棍，长度之和为 $3+4+5=12$，长度最大值为 5；
4. 选择第 $1,2,3,4$ 根小木棍，长度之和为 $1+2+3+4=10$，长度最大值为 4；
5. 选择第 $1,2,3,5$ 根小木棍，长度之和为 $1+2+3+5=11$，长度最大值为 5；
6. 选择第 $1,2,4,5$ 根小木棍，长度之和为 $1+2+4+5=12$，长度最大值为 5；
7. 选择第 $1,3,4,5$ 根小木棍，长度之和为 $1+3+4+5=13$，长度最大值为 5；
8. 选择第 $2,3,4,5$ 根小木棍，长度之和为 $2+3+4+5=14$，长度最大值为 5；
9. 选择第 $1,2,3,4,5$ 根小木棍，长度之和为 $1+2+3+4+5=15$，长度最大值为 5。

【样例 2 输入】

```
5
2 2 3 8 10
```

【样例 2 输出】

```
6
```

共有以下 6 种选择小木棍的方案，使得选出的小木棍能够拼成一个多边形：

1. 选择第 $1,2,3$ 根小木棍，长度之和为 $2+2+3=7$，长度最大值为 3；
2. 选择第 $3,4,5$ 根小木棍，长度之和为 $3+8+10=21$，长度最大值为 10；
3. 选择第 $1,2,4,5$ 根小木棍，长度之和为 $2+2+8+10=22$，长度最大值为 10；
4. 选择第 $1,3,4,5$ 根小木棍，长度之和为 $2+3+8+10=23$，长度最大值为 10；
5. 选择第 $2,3,4,5$ 根小木棍，长度之和为 $2+3+8+10=23$，长度最大值为 10；
6. 选择第 $1,2,3,4,5$ 根小木棍，长度之和为 $2+2+3+8+10=25$，长度最大值为 10。

【样例 3】

见选手目录下的 ***polygon/polygon3.in*** 与 ***polygon/polygon3.ans***。

该样例满足测试点 $7\sim10$ 的约束条件。

【样例 4】

见选手目录下的 ***polygon/polygon4.in*** 与 ***polygon/polygon4.ans***。

该样例满足测试点 $11\sim14$ 的约束条件。

【子任务】

对于所有测试数据，保证：

- $3 \leq n \leq 5,000$；
- 对于所有 $1 \leq i \leq n$，均有 $1 \leq a_i \leq 5,000$。

测试点编号	$n \leq$	$\max_{i=1}^{n} a_i \leq$
$1 \sim 3$	3	10
$4 \sim 6$	10	10^2
$7 \sim 10$	20	
$11 \sim 14$	500	
$15 \sim 17$		1
$18 \sim 20$	5,000	
$21 \sim 25$		5,000

2025 CCF 非专业级软件能力认证

CSP-J/S 2025 第二轮认证

提高级

时间：2025 年 11 月 1 日 14:30 ～ 18:30

题目名称	社团招新	道路修复	谐音替换	员工招聘
题目类型	传统型	传统型	传统型	传统型
目录	`club`	`road`	`replace`	`employ`
可执行文件名	`club`	`road`	`replace`	`employ`
输入文件名	`club.in`	`road.in`	`replace.in`	`employ.in`
输出文件名	`club.out`	`road.out`	`replace.out`	`employ.out`
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	2048 MiB	512 MiB
测试点数目	20	25	20	25
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	`club.cpp`	`road.cpp`	`replace.cpp`	`employ.cpp`

编译选项

对于 C++ 语言	`-O2 -std=c++14 -static`

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. `main` 函数的返回值类型必须是 `int`，程序正常结束时的返回值必须是 0。
3. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末换行）。
4. 选手提交的程序源文件大小不得超过 100 KiB。
5. 提交的程序源文件的放置位置请参考各省的具体要求。
6. 程序可使用的栈空间内存限制与题目的内存限制一致。
7. 禁止在源代码中改变编译器参数（如使用 `#pragma` 命令），禁止使用系统结构相关指令（如内联汇编）或其他可能造成不公平的方法。
8. 因违反上述规定而出现的问题，申诉时一律不予受理。
9. 只提供 Linux 格式附加样例文件。
10. 全国统一评测时采用的机器配置为：Intel Core Ultra 9 285K CPU @ 3.70 GHz（关闭睿频与能效核），内存 96 GB。上述时限以此配置为准。
11. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

社团招新（club）

【题目描述】

小 L 是学校算法协会的成员。在今年的学校社团招新中，小 L 一共招收了 n 个新成员，其中 n 为**偶数**。现在小 L 希望将他们分到协会不同的部门。

算法协会共设有三个部门，其中第 i $(1 \le i \le n)$ 个新成员对第 j $(1 \le j \le 3)$ 个部门的满意度为 $a_{i,j}$。定义一个分配方案的满意度为所有新成员对分配到的部门的满意度之和，也就是说，若将第 i $(1 \le i \le n)$ 个新成员分配到了第 $d_i \in \{1,2,3\}$ 个部门，则该分配方案的满意度为 $\sum_{i=1}^{n} a_{i,d_i}$。

小 L 不希望某一个部门的新成员数量过多。具体地，他要求在分配方案中，**不存在**一个部门被分配**多于** $\frac{n}{2}$ 个新成员。你需要帮助小 L 求出，满足他要求的分配方案的满意度的最大值。

【输入格式】

从文件 ***club.in*** 中读入数据。

本题包含多组测试数据。

输入的第一行包含一个正整数 t，表示测试数据组数。

接下来依次输入每组测试数据，对于每组测试数据：

- 第一行包含一个正整数 n，表示新成员的数量。
- 第 $i+1$ $(1 \le i \le n)$ 行包含三个非负整数 $a_{i,1}, a_{i,2}, a_{i,3}$，分别表示第 i 个新成员对第 $1,2,3$ 个部门的满意度。

【输出格式】

输出到文件 ***club.out*** 中。

对于每组测试数据，输出一行一个非负整数，表示满足小 L 要求的分配方案的满意度的最大值。

【样例 1 输入】

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

```
0 1 0
0 1 0
0 2 0
0 2 0
2
10 9 8
4 0 0
```

【样例 1 输出】

```
18
4
13
```

【样例 1 解释】

该样例共包含三组测试数据。

对于第一组测试数据，可以将四个新成员分别分配到第 $1,3,1,2$ 个部门，则三个部门的新成员数量分别为 $2,1,1$，均不超过 $\frac{4}{2}=2$，满意度为 $4+4+5+5=18$。

对于第二组测试数据，可以将四个新成员分别分配到第 $1,1,2,2$ 个部门，则三个部门的新成员数量分别为 $2,2,0$，均不超过 $\frac{4}{2}=2$，满意度为 $0+0+2+2=4$。

对于第三组测试数据，可以将两个新成员分别分配到第 $2,1$ 个部门，则三个部门的新成员数量分别为 $1,1,0$，均不超过 $\frac{2}{2}=1$，满意度为 $9+4=13$。

【样例 2】

见选手目录下的 ***club/club2.in*** 与 ***club/club2.ans***。

该样例满足测试点 $3,4$ 的约束条件。

【样例 3】

见选手目录下的 ***club/club3.in*** 与 ***club/club3.ans***。

该样例满足测试点 $5\sim8$ 的约束条件。

【样例 4】

见选手目录下的 ***club/club4.in*** 与 ***club/club4.ans***。

该样例满足测试点 9 的约束条件。

【样例 5】

见选手目录下的 ***club/club5.in*** 与 ***club/club5.ans***。
该样例满足测试点 15, 16 的约束条件。

【数据范围】

对于所有测试数据，保证：

- $1 \le t \le 5$；
- $2 \le n \le 10^5$，且 n 为偶数；
- 对于所有 $1 \le i \le n$，$1 \le j \le 3$，均有 $0 \le a_{i,j} \le 2 \times 10^4$。

测试点编号	$n =$	特殊性质
1	2	无
2	4	
3, 4	10	
5 ∼ 8	30	
9	200	B
10, 11		无
12	10^5	A
13, 14		B
15, 16		C
17 ∼ 20		无

特殊性质 A：对于所有 $1 \le i \le n$，均有 $a_{i,2} = a_{i,3} = 0$。
特殊性质 B：对于所有 $1 \le i \le n$，均有 $a_{i,3} = 0$。
特殊性质 C：对于所有 $1 \le i \le n$，$1 \le j \le 3$，$a_{i,j}$ 均在 $[0, 2 \times 10^4]$ 中独立均匀随机生成。

道路修复（road）

【题目描述】

C 国的交通系统由 n 座城市与 m 条连接两座城市的双向道路构成，第 $i\,(1 \le i \le m)$ 条道路连接城市 u_i 和 v_i。**任意两座城市都能通过若干条道路相互到达。**

然而，近期由于一场大地震，所有 m 条道路都被破坏了，修复第 $i\,(1 \le i \le m)$ 条道路的费用为 w_i。与此同时，C 国还有 k 个准备进行城市化改造的乡镇。对于第 $j\,(1 \le j \le k)$ 个乡镇，C 国对其进行城市化改造的费用为 c_j。在城市化改造完第 $j\,(1 \le j \le k)$ 个乡镇后，可以在这个乡镇与原来的 n 座城市间建造若干条道路，其中在它与第 $i\,(1 \le i \le n)$ 座城市间建造一条道路的费用为 $a_{j,i}$。C 国可以在这 k 个乡镇中选择**任意多个**进行城市化改造，也可以不选择任何乡镇进行城市化改造。

为尽快恢复城市间的交通，C 国政府希望以最低的费用将**原有**的 n 座城市两两连通，也即任意两座原有的城市都能通过若干条修复或新建造的道路相互到达。你需要帮助他们求出，将原有的 n 座城市两两连通的最小费用。

【输入格式】

从文件 ***road.in*** 中读入数据。

输入的第一行包含三个非负整数 n, m, k，分别表示原有的城市数量、道路数量和准备进行城市化改造的乡镇数量。

输入的第 $i+1\,(1 \le i \le m)$ 行包含三个非负整数 u_i, v_i, w_i，表示第 i 条道路连接的两座城市与修复该道路的费用。

输入的第 $j+m+1\,(1 \le j \le k)$ 行包含 $n+1$ 个非负整数 $c_j, a_{j,1}, a_{j,2}, \ldots, a_{j,n}$，分别表示将第 j 个乡镇进行城市化改造的费用与在该乡镇与原有的城市间建造道路的费用。

【输出格式】

输出到文件 ***road.out*** 中。

输出一行一个非负整数，表示将原有的 n 座城市两两连通的最小费用。

【样例 1 输入】

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

```
100 1 3 2 4
```

【样例 1 输出】

```
13
```

【样例 1 解释】

C 国政府可以选择修复第 3 条和第 4 条道路，然后将第 1 个乡镇进行城市化改造，并建造它与第 1,3 座城市间的道路，总费用为 $5+4+1+1+2=13$。可以证明，不存在比 13 更小的费用能使原有的 4 座城市两两连通。

【样例 2】

见选手目录下的 ***road/road2.in*** 与 ***road/road2.ans***。
该样例满足测试点 11,12 的约束条件。

【样例 3】

见选手目录下的 ***road/road3.in*** 与 ***road/road3.ans***。
该样例满足测试点 13,14 的约束条件。

【样例 4】

见选手目录下的 ***road/road4.in*** 与 ***road/road4.ans***。
该样例满足测试点 15,16 的约束条件。

【数据范围】

对于所有测试数据，保证：

- $1 \le n \le 10^4$，$1 \le m \le 10^6$，$0 \le k \le 10$；
- 对于所有 $1 \le i \le m$，均有 $1 \le u_i, v_i \le n$，$u_i \ne v_i$ 且 $0 \le w_i \le 10^9$；
- 对于所有 $1 \le j \le k$，均有 $0 \le c_j \le 10^9$；
- 对于所有 $1 \le j \le k$，$1 \le i \le n$，均有 $0 \le a_{j,i} \le 10^9$；
- 任意两座原有的城市都能通过若干条原有的道路相互到达。

测试点编号	$n \leq$	$m \leq$	$k \leq$	特殊性质
$1 \sim 4$	10^4	10^6	0	无
$5, 6$	10^3	10^5	5	A
$7, 8$				无
$9, 10$		10^6		A
$11, 12$				无
$13, 14$			10	A
$15, 16$				无
$17, 18$	10^4		5	A
$19, 20$				无
$21 \sim 25$			10	

特殊性质 A：对于所有 $1 \leq j \leq k$，均有 $c_j = 0$ 且均存在 $1 \leq i \leq n$ 满足 $a_{j,i} = 0$。

谐音替换（replace）

【题目描述】

小 W 是一名喜欢语言学的算法竞赛选手。在语言学中，谐音替换是指将原有的字词替换为读音相同或相近的字词。小 W 发现，谐音替换的过程可以用字符串来进行描述。具体地，小 W 将谐音替换定义为以下字符串问题：

给定 n 个字符串二元组，第 i $(1 \le i \le n)$ 个字符串二元组为 $(s_{i,1}, s_{i,2})$，满足 $|s_{i,1}| = |s_{i,2}|$，其中 $|s|$ 表示字符串 s 的长度。

对于字符串 s，定义 s 的**替换**如下：

- 对于 s 的某个子串 y，若存在 $1 \le i \le n$ 满足 $y = s_{i,1}$，则将 y 替换为 $y' = s_{i,2}$。具体地，设 $s = x + y + z$，其中 x 和 z 可以为空，“+”表示字符串拼接，则 s 的替换将得到字符串 $s' = x + y' + z$。

小 W 提出了 q 个问题，第 j $(1 \le j \le m)$ 个问题会给定两个**不同**的字符串 $t_{j,1}, t_{j,2}$，她想知道有多少种字符串 $t_{j,1}$ 的替换能够得到字符串 $t_{j,2}$。两种 s 的替换不同当且仅当**子串 y 的位置不同或用于替换的二元组 $(s_{i,1}, s_{i,2})$ 不同**，即 x, z 不同或 i 不同。你需要回答小 W 提出的所有问题。

【输入格式】

从文件 ***replace.in*** 中读入数据。

输入的第一行包含两个正整数 n, q，分别表示字符串二元组的数量和小 W 提出的问题的数量。

输入的第 $i + 1$ $(1 \le i \le n)$ 行包含两个字符串 $s_{i,1}, s_{i,2}$，表示第 i 个字符串二元组。

输入的第 $j + n + 1$ $(1 \le j \le q)$ 行包含两个字符串 $t_{j,1}, t_{j,2}$，表示小 W 提出的第 j 个问题。

【输出格式】

输出到文件 ***replace.out*** 中。

输出 q 行，其中第 j $(1 \le j \le q)$ 行包含一个非负整数，表示替换后得到字符串 $t_{j,2}$ 的字符串 $t_{j,1}$ 的替换的数量。

【样例 1 输入】

```
4 2
xabcx xadex
ab cd
bc de
```

```
aa bb
xabcx xadex
aaaa bbbb
```

【样例 1 输出】

```
2
0
```

【样例 1 解释】

对于小 W 的第一个询问，共有 2 种 $t_{1,1}$ 的替换能够得到 $t_{1,2}$：

1. 令 x, z 均为空串，$y = \texttt{xabcx}$，$i = 1$，则 $y' = \texttt{xadex}$，替换后得到 xadex；
2. 令 $x = \texttt{xa}$，$y = \texttt{bc}$，$z = \texttt{x}$，$i = 3$，则 $y' = \texttt{de}$，替换后得到 xadex。

【样例 2 输入】

```
3 4
a b
b c
c d
aa bb
aa b
a c
b a
```

【样例 2 输出】

```
0
0
0
0
```

【样例 3】

见选手目录下的 ***replace/replace3.in*** 与 ***replace/replace3.ans***。

该样例满足测试点 11, 12 的约束条件。

【样例 4】

见选手目录下的 ***replace/replace4.in*** 与 ***replace/replace4.ans***。
该样例满足测试点 15, 16 的约束条件。

【数据范围】

设 $L_1 = \sum_{i=1}^{n} |s_{i,1}| + |s_{i,2}|$，$L_2 = \sum_{j=1}^{q} |t_{j,1}| + |t_{j,2}|$。对于所有测试数据，保证：

- $1 \le n, q \le 2 \times 10^5$；
- $2 \le L_1, L_2 \le 5 \times 10^6$；
- 对于所有 $1 \le i \le n$，$s_{i,1}, s_{i,2}$ 均仅包含小写英文字母，且 $|s_{i,1}| = |s_{i,2}|$；
- 对于所有 $1 \le j \le q$，$t_{j,1}, t_{j,2}$ 均仅包含小写英文字母，且 $t_{j,1} \ne t_{j,2}$。

测试点编号	$n, q \le$	$L_1, L_2 \le$	特殊性质
$1, 2$	10^2	200	无
$3 \sim 5$	10^3	$2,000$	
6		10^6	AB
$7, 8$	10^4		A
$9, 10$	2×10^5		B
$11, 12$		2×10^6	无
$13, 14$		5×10^6	A
$15, 16$			B
$17 \sim 20$			无

特殊性质 A：$q = 1$。

特殊性质 B：定义字符串 s 为**特别的**，当且仅当字符串 s 仅包含字符 `a` 和 `b`，且字符 `b` 在 s 中出现**恰好**一次。对于所有 $1 \le i \le n$，$s_{i,1}, s_{i,2}$ 均为特别的，且对于所有 $1 \le j \le q$，$t_{j,1}, t_{j,2}$ 均为特别的。

员工招聘（employ）

【题目描述】

小 Z 和小 H 想要合伙开一家公司，共有 n 人前来应聘，编号为 $1 \sim n$。小 Z 和小 H 希望录用至少 m 人。

小 H 是面试官，将在接下来 n 天每天面试一个人。小 Z 负责决定应聘人前来面试的顺序。具体地，小 Z 可以选择一个 $1 \sim n$ 的排列 p，然后在第 i $(1 \le i \le n)$ 天通知编号为 p_i 的人前来面试。

小 H 准备了 n 套难度不一的面试题。由于 n 个前来应聘的人水平大致相同，因此对于同一套题，所有人的作答结果是一致的。具体地，第 i $(1 \le i \le n)$ 天的面试题的难度为 $s_i \in \{0, 1\}$，其中 $s_i = 0$ 表示这套题的难度较高，没有人能够做出；$s_i = 1$ 表示这套题的难度较低，所有人均能做出。小 H 会根据面试者的作答结果决定是否录用，即如果面试者没有做出面试题，则会拒绝，否则会录用。

然而，每个人的耐心都有一定的上限，如果在他面试之前未录用的人数过多，则他会直接放弃参加面试。具体地，编号为 i $(1 \le i \le n)$ 的人的耐心上限可以用非负整数 c_i 描述，若在他之前已经有不少于 c_i 人被拒绝或放弃参加面试，则他也将放弃参加面试。

小 Z 想知道一共有多少种面试的顺序 p 能够让他们录用至少 m 人。你需要帮助小 Z 求出，能够录用至少 m 人的排列 p 的数量。由于答案可能较大，你只需要求出答案对 $998,244,353$ 取模后的结果。

【输入格式】

从文件 ***employ.in*** 中读入数据。

输入的第一行包含两个正整数 n, m，分别表示前来应聘的人数和希望录用的人数。

输入的第二行包含一个长度为 n 的字符串 $s_1 \dots s_n$，表示每一天的面试题的难度。

输入的第三行包含 n 个非负整数 $c_1, c_2, \dots, c_n$，表示每个人的耐心上限。

【输出格式】

输出到文件 ***employ.out*** 中。

输出一行一个非负整数，表示能够录用至少 m 人的排列 p 的数量对 $998,244,353$ 取模后的结果。

【样例 1 输入】

```
3 2
101
1 1 2
```

【样例 1 输出】

```
2
```

【样例 1 解释】

共有以下 2 种面试的顺序 p 能够让小 Z 和小 H 录用至少 2 人：

1. $p = [1, 2, 3]$，依次录用编号为 1 的人和编号为 3 的人；
2. $p = [2, 1, 3]$，依次录用编号为 2 的人和编号为 3 的人。

【样例 2 输入】

```
10 5
1101111011
6 0 4 2 1 2 5 4 3 3
```

【样例 2 输出】

```
2204128
```

【样例 3】

见选手目录下的 ***employ/employ3.in*** 与 ***employ/employ3.ans***。
该样例满足测试点 6 ∼ 8 的约束条件。

【样例 4】

见选手目录下的 ***employ/employ4.in*** 与 ***employ/employ4.ans***。
该样例满足测试点 12 ∼ 14 的约束条件。

【样例 5】

见选手目录下的 ***employ/employ5.in*** 与 ***employ/employ5.ans***。
该样例满足测试点 18 ∼ 21 的约束条件。

【数据范围】

对于所有测试数据，保证：

- $1 \le m \le n \le 500$；
- 对于所有 $1 \le i \le n$，均有 $s_i \in \{0, 1\}$；
- 对于所有 $1 \le i \le n$，均有 $0 \le c_i \le n$。

测试点编号	$n \le$	m	特殊性质
$1, 2$	10	$\le n$	无
$3 \sim 5$	18		
$6 \sim 8$	10^2		A
$9 \sim 11$			无
$12 \sim 14$	500	$= 1$	
15		$= n$	
$16, 17$		$\le n$	A
$18 \sim 21$			B
$22 \sim 25$			无

特殊性质 A：对于所有 $1 \le i \le n$，均有 $s_i = 1$。

特殊性质 B：在 $s_1, s_2, \ldots, s_n$ 中最多只有 18 个取值为 1，即 $\sum_{i=1}^{n} s_i \le 18$。