

2025 CCF 非专业软件能力认证

CSP-J/S 2025 第二轮认证

时间：2025 年 10 月 25 日 8:30 ~ 12:00

题目名称	压缩文章	序列	最小公倍数	旅游
题目类型	传统题	传统题	传统题	传统题
目录	str	seq	lcm	graph
可执行文件名	str	seq	lcm	graph
输入文件名	str.in	seq.in	lcm.in	graph.in
输出文件名	str.out	seq.out	lcm.out	graph.out
测试点时限	1.0 秒	1.0 秒	1.0 秒	3.0 秒
内存限制	512 MB	512 MB	512 MB	1024 MB
子任务数目	3	3	10	20
是否捆绑测试	是	是	否	否

提交源程序文件名

对于 C++	str.cpp	seq.cpp	lcm.cpp	graph.cpp
--------	---------	---------	---------	-----------

编译选项

对于 C++	-lm -std=c++14 -O2 -Wl,--stack=998244353
--------	--

注意事项与提醒（请选手务必仔细阅读）

1. 选手提交的源程序必须存放在已建立好的，且带有**样例文件**和**下发文件**的文件夹中，文件名称与对应试题英文名一致；
2. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
3. C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 0。
4. **对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。**
5. 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**。
6. 程序可使用的栈空间大小与该题内存空间限制一致。
7. 在终端中执行命令 `ulimit -s unlimited` 可将当前终端下的栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
8. 若无特殊说明，每道题的**代码大小限制为 100KB**。
9. 若无特殊说明，输入与输出中同一行的相邻整数、字符串等均使用一个空格分隔。
10. 输入文件中可能存在行末空格，请选手使用更完善的读入方式（例如 `scanf` 函数）避免出错。

11. 直接复制 PDF 题面中的多行样例，数据将带有行号，建议选手直接使用对应目录下的样例文件进行测试。
12. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
13. 请务必使用题面中规定的的编译参数，保证你的程序在本机能够通过编译。此外**不允许在程序中手动开启其他编译选项**，一经发现，本题成绩以 0 分处理。

压缩文章 (str)

【题目描述】

i18n（其来源是英文单词 internationalization 的首末字符 i 和 n，18 为中间的字符数）是“国际化”的简称。

翁老师觉得这样的压缩方法很好，于是找到了一篇仅包含大小写英文字母、空格和英文句号 . 的英文短文，请你用这样的方法帮他把短文压缩一下，具体要求如下：

- 空格及句号保持原样输出。
- 单词由空格或句号分隔。
- 每个单词都按照 i18n 的形式输出，具体要求如下：
 - 如果小于等于两个字符，保持原样输出。
 - 如果大于两个字符，第一个字符和最后一个字符保持原样输出，中间输出一个整数，即这个中间的字符数量。

【输入格式】

一行，一个待压缩的字符串：s。

【输出格式】

一行，一个压缩好之后的字符串。

【样例 1 输入】

```
1 includE bits stdc.h using namespace std string s t
```

【样例 1 输出】

```
1 i5E b2s s2c.h u3g n7e s1d s4g s t
```

注意 stdc.h 是一个由 . 隔开的两个单词分别是 stdc 和 h。其中 stdc 会被压缩为 s2c

【样例 2 输入】

```
1 abc
```

【样例 2 输出】

```
1 a1c
```

【样例 3】

见选手目录下的 str/str3.in 与 str/str3.ans

【数据范围与提示】

对于 100% 的数据， $1 \leq |s| \leq 1000$ ，保证至少包含一个大小写英文字母。
本题采用捆绑测试

子任务编号	特殊限制	分值
1	s 中只包含一个英语单词。	30
2	s 中不存在句号 .，并且所有英语单词之间都恰好一个空格。	30
3	无	40

序列 (seq)

【题目描述】

翁老师给你一个长度为 n 的数列，其中第 i 项为 i ，即整个数列为 $1 \sim n$ 。

$n = 8$ 时数列为：1, 2, 3, 4, 5, 6, 7, 8

现在告诉你两个整数 a, b ，翁老师对这个数列进行了 a 次如下操作：

- 将序列的前 b 项放到序列末尾。其余元素顺序保持不变。

求这个序列最终的样子。

【输入格式】

第一行，输入三个整数 n, a, b 。

【输出格式】

输出一行空格隔开的 n 个整数，即序列最终的样子。

【样例 1 输入】

1 8 1 3

【样例 1 输出】

1 4 5 6 7 8 1 2 3

【样例 1 解释】

进行了 1 次，把前 3 项放到序列末尾。

【样例 2 输入】

1 3 2 2

【样例 2 输出】

1 2 3 1

【样例 2 解释】

变化过程如下：

- 1, 2, 3 (初始)
- 3, 1, 2 (1 次操作后)
- 2, 3, 1 (2 次操作后)

【样例 3】

见选手目录下的 seq/seq3.in 与 seq/seq3.ans

【数据范围与提示】

对于 100% 的数据， $1 \leq b \leq n \leq 1000$ ， $1 \leq a \leq 10^{12}$ 。

本题采用捆绑测试

子任务编号	特殊限制	分值
1	$a = 1$	30
2	$a \leq 100$	30
3	无特殊限制	40

最小公倍数 (lcm)

【题目描述】

在遥远的数学王国里，有一座被智慧之光笼罩的山峰——数论峰。这座山峰上居住着各式各样的数学精灵，它们以解开数字的奥秘为乐。

数论峰中一共有 m 只数论精灵，每只精灵每天都会选择一个权值为 $[1, n]$ 中数字作为一天的幸运数，不同的精灵可以选择相同的数字作为幸运数。

当一天中所有数论精灵的幸运数的最小公倍数恰好为 n 时，数论峰中的古老大门将打开。

聪明的你想知道，有多少方案使得所有数论精灵的幸运数的最小公倍数恰好为 n ，答案对 998244353 取模。

两种方案不同当且仅当有一只精灵的幸运数不同。

【输入格式】

本题共有 T 组数据。

输入第一行，包含一个正整数，表示 T 。

之后对于每组数据，输入 1 行，包含 2 个正整数 n, m 。

【输出格式】

输出共 T 行，每行输出 1 个整数，表示最终答案，答案对 998244353 取模。

【样例 1 输入】

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

【样例 1 输出】

1	9
2	15
3	63
4	255

【样例 1 解释】

当 $n = 6, m = 2$ 时,合法的解有 $(2, 3), (3, 2), (1, 6), (6, 1), (2, 6), (3, 6), (6, 2), (6, 3), (6, 6)$ 。

【样例 2 输入】

```
1 5
2 4 5
3 5 4
4 98 97
5 100000 9982443
6 123454 2
```

【样例 2 输出】

```
1 211
2 15
3 306665342
4 947591178
5 27
```

【样例 3】

见选手目录下的 lcm/lcm3.in 与 lcm/lcm3.ans

【样例 4】

见选手目录下的 lcm/lcm4.in 与 lcm/lcm4.ans

【数据范围与提示】

- 对于 20% 的数据, $1 \leq n, m \leq 5, 1 \leq T \leq 10$ 。
- 对于 40% 的数据, $1 \leq n, m \leq 500, 1 \leq T \leq 10$ 。
- 对于另外 10% 的数据, n 为质数。
- 对于另外 30% 的数据, $m = 2$ 。
- 对于 100% 的数据, $1 \leq T \leq 10^3, 1 \leq n, m \leq 10^9$

旅游 (graph)

【题目描述】

n 个点 m 条边的有权无向图，每条边有一个开放时间 $[s, t]$ ，只有在开放时间内，你才可以走上这条路，并在路径长度个单位时间抵达终点。（若截止时间到达时仍在该路径上，仍可继续行走至终点。）

可以进行 k 次修改一个边的一个时间，问最快从 1 到达 n 时间。（出发的时间是 0，如果到达某个点时还没到某条边的开放时间，除了直接修改开放时间之外，你也可以在这个点原地休息，直到开放时间再出发。）

【输入格式】

本题有 T 组测试数据。

对于每一组测试数据，第一行三个数 n, m, k 。

接下来 m 行，每行五个整数 x_i, y_i, s_i, t_i, w_i 表示第 i 条边的两个端点, 开放时间, 关闭时间, 路径长度。

【输出格式】

对于每组数据，输出一行答案，保证最终可以到点 n 。

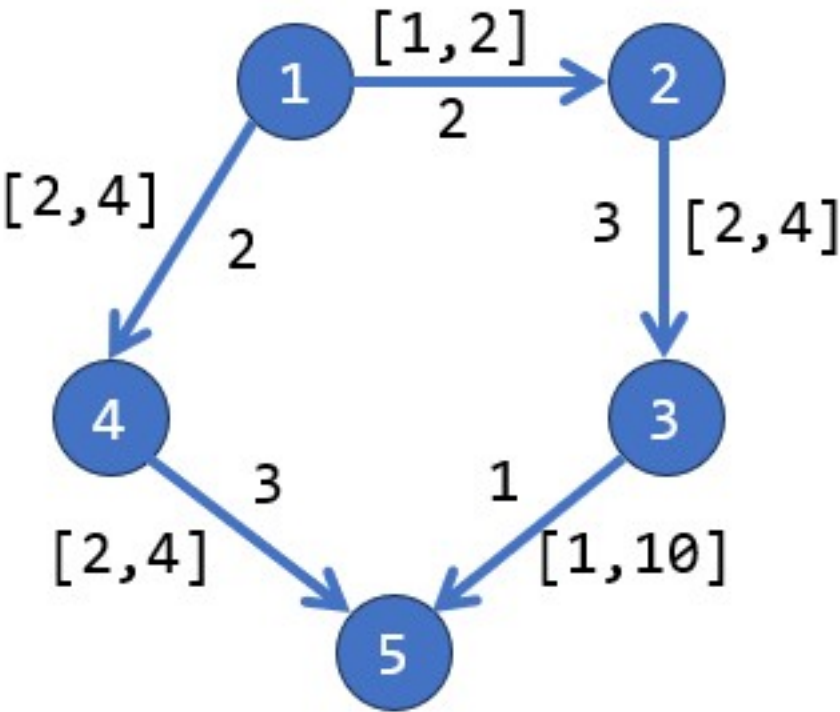
【样例 1 输入】

1	2
2	5 5 1
3	1 2 1 2 2
4	2 3 2 4 3
5	3 5 1 10 1
6	4 5 2 4 3
7	1 4 2 4 2
8	5 5 0
9	1 2 1 2 2
10	2 3 2 4 3
11	3 5 1 10 1
12	4 5 2 4 3
13	1 4 2 4 2

【样例 1 输出】

```
1 5
2 7
```

【样例 1 解释】



样例的图如上（注意本题为无向图，箭头只是为了指示下面样例解释中的两条路径）
可以把 1 到 4 这条有向边的开放时间的起始时间改为 0，即可通过 1, 4, 5 这条路径，这样修改一次，花费 5 的时间可以到达终点。
如果不允许修改。可以在起点等待到时间 2 再出发，沿着 1, 4, 5 到达终点；也可以在起点等待到时间 1 再出发，沿着 1, 2, 3, 5 到达终点。两种方案到达终点的时间都是 7。

【样例 2】

见选手目录下的 graph/graph2.in 与 graph/graph2.ans

【样例 3】

见选手目录下的 graph/graph3.in 与 graph/graph3.ans

【数据范围与提示】

- 对于 20% 的数据, $2 \leq n \leq 12$, $0 \leq m \leq 30$, $0 \leq k \leq 1$, $1 \leq T \leq 2$ 。
- 对于 50% 的数据, $2 \leq n \leq 1002$, $0 \leq m \leq 2000$, $0 \leq k \leq 30$ 。
- 对于 100% 的数据, $2 \leq n \leq 30002$, $0 \leq m \leq 60000$, $0 \leq k \leq 30$, $0 \leq s, t, w \leq 10^9$, $1 \leq T \leq 5$ 。