

2025 0101 CCF 非专业级软件能力认证

2025 0101 CCF 非专业级软件能力认证

CSP-J/S 2025 第二轮模拟认证

时间：2025 年 10月5日 10:30 ~ 12:00

题目名称	子串取模	前后洗牌	序列重排	划分	蛋糕	矩阵
题目类型	传统型	传统型	传统型	传统型	传统型	传统型
目录	modulo	shuffle	arrange	partition	cake	matrix
可执行文件名	modulo	shuffle	arrange	partition	cake	matrix
输入文件名	modulo.in	shuffle.in	arrange.in	partition.in	cake.in	matrix.in
输出文件名	modulo.out	shuffle.out	arrange.out	partition.out	cake.out	matrix.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	256 MiB	256 MiB	256 MiB	256 MiB	256 MiB	256 MiB
测试点数目	20	15	10	10	10	10
测试点是否等分	是	是	是	是	是	是

提交源程序文件名

| 对于C++语言 | modulo.cpp | shuffle.cpp | arrange.cpp | partition.cpp | cake.cpp | matrix.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. 提交的程序代码文件的放置位置请参考各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
5. 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回车）。
6. 选手提交的程序源文件必须不大于 100KB。
7. 程序可使用的栈空间内存限制与题目的内存限制一致。
8. 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @3.70GHz，内存 32GB。
9. 只提供 Linux 格式附加样例文件。
10. 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。

题目内容

子串取模 (modulo)

题目描述

给你一个数字构成的字符串 s 。有 n 个问题。

每个问题会给你三个整数 l, r, m ，请输出 s 从左数第 l 个字符到第 r 个字符构成的数除以 m 的余数。

输入格式

第一行一个整数 n 。

第二行一个字符串 s 。

接下来 n 行，每行两个整数 l, r, m ，即当前问题。

输出格式

n 行，每行分别是每个问题的答案。

```
3
12345678901234567890
9 15 33
4 7 100
1 2 1000
```

```
12
67
12
```

样例解释

```
9012345 % 33 = 12
```

```
4567 % 100 = 67
```

```
12 % 1000 = 12
```

数据规模与约定

对于 100% 的数据, $1 \leq n, |s| \leq 1000, 1 \leq l \leq r \leq |s|, 1 \leq m \leq 998244353$ 。

- 子任务 1 (30 分) : 保证 $l = r$ 。
- 子任务 2 (30 分) : 保证 $m = 1000$ 。
- 子任务 3 (40 分) : 保证 $m = 998244353$ 。

题目内容

前后洗牌 (shuffle)

题目描述

给你一个字符串 s , 请进行一轮“洗牌”, 输出“洗牌”后的字符串。

“洗牌”时会构建一个新的字符串 t , t 初始为一个空字符串, 然后会重复执行以下步骤直到 s 为空:

1. 如果 s 还有字符, 把 s 最前面的字符添加到 t 的末尾, 然后把 s 最前面的字符删掉。
2. 如果 s 还有字符, 把 s 最后面的字符添加到 t 的末尾, 然后把 s 最后面的字符删掉。

s 中的字符都被删完后, t 就是“洗牌”后的字符串。

输入格式

输入一个字符串 s 。

输出格式

输出“洗牌”后的字符串。

```
abcdefg
```

```
agbfced
```

数据规模与约定

对于 100% 的数据, s 中仅包含小写英文字母, $1 \leq |s| \leq 100$, $|s|$ 表示字符串 s 的长度。

- 子任务 1 (30 分) : 保证 s 中全是小写英文字母 `a`。
- 子任务 2 (30 分) : 保证 $|s|$ 是一个偶数。

- 子任务 3 (40 分) : 没有特殊限制。

题目内容

序列重排 (arrange)

序列重排 (arrange)

题目描述

小 C 有一个长度为 n 的序列 A 。

小 K 定义一个序列 A 的权值为 $\text{mex}\{A_1 + A_2, A_2 + A_3, \dots, A_{n-1} + A_n\}$, 其中 $\text{mex}\{S\}$ 表示集合 S 中最小的未出现的非负整数。

小 C 现在可以将序列 A 任意排列, 他想要让序列 A 的权值尽可能小, 你能告诉他该最小权值吗?

输入格式

输入的第一行包含一个整数 n 。

接下来一行包含 n 个整数, 第 i 个整数表示 A_i 。

输出格式

输出共一行, 包含一个整数, 表示最小权值。

样例 1 输入

```
3
0 0 1
```

样例 1 输出

```
0
```

样例 1 解释

将序列 A 重排为 $A_1 = 0, A_2 = 1, A_3 = 0$, 可以得到最小权值 0。

样例 2 输入

```
5
0 1 2 3 2
```

样例 2 输出

```
0
```

其余样例见下发文件。

数据规模与约定

- 对于 40% 的数据，保证 $n \leq 10$ 。
- 对于另外 20% 的数据，保证序列 A 中 0 的个数不超过 $\lfloor \frac{n+1}{2} \rfloor$ 。
- 对于 100% 的数据， $2 \leq n \leq 10^6$ ， $0 \leq A_i \leq 10^9$ 。

题目内容

划分 (partition)

划分 (partition)

题目描述

小 C 有一个长度为 n 的序列 A ，现在他想把序列 A 划分成至少 2 段。

小 C 定义 l_i, r_i 表示划分出的第 i 个子段的左右端点，这个子段的子段和 $b_i = \sum_{j=l_i}^{r_i} A_j$ 。

小 C 认为一个合法的划分需要满足 $l_i \leq r_i$ ，且对于 $\forall 1 \leq j < k$ ，有 $l_{j+1} = r_j + 1$ ，并且 $l_1 = 1, r_k = n$ 。(假设划分出了 k 段)

小 C 想要求一种划分方案使得 $\gcd(b_1, b_2, \dots, b_k)$ 最大，你能告诉他该最大值吗？

输入格式

输入的第一行包含一个整数 n 。

接下来一行包含 n 个整数，第 i 个整数表示 A_i 。

输出格式

输出共一行，包含一个整数，表示 $\gcd(b_1, b_2, \dots, b_k)$ 的最大值。

样例 1 输入

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

样例 1 输出

```
3
```

样例 2 输入

```
6
7 7 7 7 7 7
```

样例 2 输出

其余样例见下发文件。

数据规模与约定

- 对于 30% 的数据，保证 $n \leq 20$ 。
- 对于另 30% 的数据，保证 $n \leq 100$, $1 \leq A_i \leq 3$ 。
- 对于另 20% 的数据，保证 $A_i = 1$ 且 $2|n$ 。
- 对于 100% 的数据，保证 $1 \leq n \leq 10^5$, $1 \leq A_i \leq 10^9$ 。

题目内容

蛋糕 (cake)

题目描述

小 C 过生日了，他买了 m 块蛋糕，第 i 块蛋糕的大小为 a_i 。

小 C 喜欢 2 这个数，所以他买的所有蛋糕的大小一定是 2 的整数幂，即 $\forall 1 \leq i \leq m, \exists 0 \leq k, a_i = 2^k$ 。

小 C 可以用一把刀切蛋糕（切的蛋糕大小必须大于 1），他可以将一块大小为 x 的蛋糕一分为二切成两块大小为 $\frac{x}{2}$ 的蛋糕。

小 C 想要选出一些蛋糕送给朋友，但小 C 有强迫症，他想要选出蛋糕的大小之和为 n 。

由于切蛋糕是需要体力的，小 C 想要知道最少需要切多少次蛋糕使得他能够选出一些蛋糕使得选出蛋糕的大小之和为 n 。

如果怎样切蛋糕都无法选出一些蛋糕使得选出蛋糕的大小之和为 n ，请告诉小 C 不可能（即输出 -1 ）。

输入格式

第一行输入两个数字 n, m ，分别表示选出蛋糕的大小之和，最初蛋糕的块数。

第二行输入 m 个整数，第 i 个整数表示第 i 块蛋糕的大小 a_i 。

输出格式

共一行，输出一个整数，表示切蛋糕的最小次数，若不可能输出 -1 。

样例 1 输入

```
10 3
1 1 32
```

样例 1 输出

样例 1 解释

- 选择大小为 32 的蛋糕切一刀，变成两块大小为 16 的蛋糕。
- 选择大小为 16 的蛋糕切一刀，变成两块大小为 8 的蛋糕。

可以选出大小分别为 1, 1, 8 的蛋糕，使得大小之和为 10，故最小次数为 2。

样例 2 输入

```
20 5
1 1 2 8 16
```

样例 2 输出

```
0
```

其余样例见下发文件。

数据规模与约定

- 对于 20% 的数据，保证 $n \leq 256$, $m \leq 4$, $1 \leq a_i \leq 512$ 。
- 对于 40% 的数据，保证 $n \leq 4096$ 。
- 对于另 20% 的数据，保证 $m = 1$ 。
- 对于另 20% 的数据，保证 $m = 2$ 。
- 对于 100% 的数据，保证 $1 \leq n \leq 10^{18}$, $1 \leq m \leq 10^5$, $1 \leq a_i \leq 10^{13}$ 。

题目内容

矩阵 (matrix)

题目描述

小 C 有一个大小为 $n \times m$ 的 $\{0,1\}$ 矩阵 A 。

小 C 认为第 i 列是好的当且仅当第 i 列中 $\{1\}$ 刚好出现了一次，即 $\sum_{j=1}^n [A_{j,i} = 1] = 1$ 。

小 C 可以进行以下操作任意次：选择矩阵 A 中的某一行将其 $\{0,1\}$ 翻转（即 $\{0\}$ 变成 $\{1\}$ ， $\{1\}$ 变成 $\{0\}$ ）。

小 C 想要让矩阵 A 中好的列数尽可能多，你能告诉他这个最大值吗？

输入格式

第一行输入两个数字 n, m ，分别表示矩阵的长与宽。

接下来 n 行，每行包含一个长度为 m 仅由 $\{0,1\}$ 组成的字符串。

输出格式

共一行，输出一个整数，表示矩阵 A 中最多的好的列数。

样例 1 输入

```
3 4
0101
0110
1011
```

样例 1 输出

```
3
```

样例 1 解释

将每一行都进行翻转，矩阵 A 变为：

```
1010
1001
0100
```

此时第 2, 3, 4 列是好的，故答案为 3。

样例 2 输入

```
3 3
101
111
000
```

样例 2 输出

```
2
```

其余样例见下发文件。

数据规模与约定

- 对于 20% 的数据，保证 $n, m \leq 16$ 。
- 对于 40% 的数据，保证 $n, m \leq 100$ 。
- 对于 60% 的数据，保证 $n, m \leq 500$ 。
- 对于另 20% 的数据，保证 $n \times m \leq 70000$ 。
- 对于 100% 的数据，保证 $1 \leq n, m \leq 3 \times 10^5$, $1 \leq n \times m \leq 3 \times 10^5$ 。