

苏州市人工智能学会

青少年人工智能核心算法素养考核(SACCC)

五级

时间：2026 年 6 月 28 日 08:30 ~ 12:00

题目名称	Token 管理	矩阵中的我	数据集指派	神仙跳舞
题目类型	传统型	传统型	传统型	传统型
目录	manage	grid	change	dance
可执行文件名	manage	grid	change	dance
输入文件名	manage.in	grid.in	change.in	dance.in
输出文件名	manage.out	grid.out	change.out	dance.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	10	10	10	10
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	manage.cpp	grid.cpp	change.cpp	dance.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监考老师的具体要求。
4. 因违反以上三点而出现的错误或问题，产生的后果由本人承担。
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. 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。上述时限以此配置为准。

Token 管理 (manage)

【题目描述】

某公司部署了 n 个 AI 模型服务, 编号为 1 到 n 。最开始 (时刻 0), 第 i 个模型剩余的 Token 配额为 a_i 。

由于持续有用户调用推理接口, 模型的 Token 配额会随着时间自动消耗: 如果某个模型在某段时间内没有被充值, 那么每经过 1 单位时间, 它的 Token 配额就会减少 1。但 Token 配额最低不会降到 0 以下 (即配额耗尽后不再继续扣减)。

接下来有 q 个事件按时间顺序发生。第 j 个事件发生在时刻 t_j , 保证时间递增, 即 $0 < t_1 < t_2 < \dots < t_q$ 。

事件共有以下两种类型:

- 1 t x v: 在时刻 t , 为第 x 个模型充值 Token, 使其配额增加 v 。
- 2 t x: 在时刻 t , 查询第 x 个模型当前的剩余 Token 配额。

请按顺序处理所有的事件, 并输出所有类型为 2 的事件的答案。

【输入格式】

第一行包含两个正整数 n 和 q , 分别表示模型的数量和事件的数量。

第二行包含 n 个非负整数 $a_1, a_2, \dots, a_n$, 表示每个模型在时刻 0 的初始 Token 配额。

接下来 q 行, 每行描述一个事件:

- 若格式为 1 t x v, 表示为第 x 个模型充值 Token v ;
- 若格式为 2 t x, 表示查询第 x 个模型的剩余 Token 配额。

【输出格式】

对于每个类型为 2 的查询事件, 输出一行一个整数, 表示对应模型在当前时刻的剩余 Token 配额。

【样例 1 输入】

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

【样例 1 输出】

```
1 4
2 4
3 2
4 0
5 3
```

【样例 1 解释】

各模型的初始 Token 配额为 $[5, 0, 4]$ 。

- 2 1 1: 第 1 个模型在时刻 1 的剩余配额为 $5 - 1 = 4$, 输出 4。
- 1 2 2 6: 第 2 个模型在时刻 2 的配额仍为 0, 充值后变为 6。
- 2 4 2: 第 2 个模型距离上次充值经过 2 单位时间, 配额消耗至 4, 输出 4。
- 1 6 1 3: 第 1 个模型到时刻 6 配额已耗尽至 0, 充值后变为 3。
- 2 7 1: 第 1 个模型距离上次充值经过 1 单位时间, 配额消耗至 2, 输出 2。
- 2 9 3: 第 3 个模型从时刻 0 到时刻 9 未被充值, 配额降至 0, 输出 0。
- 1 10 3 5: 第 3 个模型在时刻 10 的配额为 0, 充值后变为 5。
- 2 12 3: 第 3 个模型距离上次充值经过 2 单位时间, 配额消耗至 3, 输出 3。

【样例 2 输入】

```
1 1 5
2 100
3 1 1 1 5
4 2 3 1
5 1 10 1 4
6 2 11 1
7 2 20 1
```

【样例 2 输出】

```
1 102
2 98
3 89
```

【数据规模与约定】

- 对于 50% 的数据: $1 \leq n, q \leq 10$
- 对于 100% 的数据: $1 \leq n, q \leq 2 \times 10^5$, $0 \leq a_i \leq 10^9$, 事件中的 t 按升序给出, 且 $1 \leq t_i \leq 10^9$, $1 \leq x \leq n$, $1 \leq v \leq 10^9$

矩阵中的我 (grid)

【题目描述】

你正在训练一个大规模多模态 AI 模型，该模型包含一个 $n \times m$ 的特征交互矩阵，其中第 i 行第 j 列的元素表示特征 i 与特征 j 的交叉注意力权重，其值为 $i \times j$ （行和列都从 1 开始编号）。

在模型推理过程中，你需要将这 $n \times m$ 个注意力权重按照非严格单调递增的顺序依次输入到排序归一化层中进行处理。

请问，你输入的第 k 个注意力权重是多少。

【输入格式】

一行，三个整数 n, m, k ，分别表示特征维度（行数）、特征数量（列数）和第 k 个元素。

【输出格式】

一行，输出第 k 个注意力权重的值。

【样例 1 输入】

1 2 2 2

【样例 1 输出】

1 2

【样例 1 解释】

此特征交互矩阵内包含 2 行 2 列,注意力权重分别是 $1 \times 1, 1 \times 2, 2 \times 1, 2 \times 2$ ，按照值的大小升序排列是 1、2、2、4，排在第 2 的是 2。

【样例 2 输入】

1 2 3 4

【样例 2 输出】

1 3

【样例 3 输入】

1 500000 500000 249999999999

【样例 3 输出】

1 249999500000

【数据范围】

- 对于 30% 的数据： $1 \leq n, m \leq 500$ 。
- 对于 100% 的数据： $1 \leq n, m \leq 5 \times 10^5$, $1 \leq k \leq n \times m$ 。

数据集指派 (change)

【题目描述】

哪吒和孙悟空正在合作标注一个包含 n 条样本的数据集，样本编号为 1 到 n 。第 i 条样本记录了当前标注负责人 S_i 和标注难度系数 P_i 。当 S_i 为 N 时，负责人是哪吒 (Nezha)；当 S_i 为 W 时，负责人是孙悟空 (Wukong)。

这里，**反转负责人**是指将 N 改为 W，将 W 改为 N。

AI 训练需要的数据集要求特定样本由特定角色标注。给定 Q 个询问。对于第 j 个询问，你可以执行以下操作至多一次（也可以选择不执行操作）：

操作：选择一对整数 (l, r) ，满足 $L_j \leq l \leq r \leq R_j$ ，并将第 l 到第 r 条样本（含两端）的所有标注负责人反转。

考虑操作后（或不执行操作时），在第 L_j 到第 R_j 条样本中，由 **孙悟空 (W)** 负责的样本的标注难度系数之和。求当操作选择最优时，这个总和的最小值。

每个询问是独立的。一个询问中的操作不会影响原始的负责人/难度系数信息，也不会影响其他询问。

【输入格式】

第 1 行： n （样本个数）和 Q （询问个数）

接下来 n 行：第 i 行包含 S_i （负责人）和 P_i （标注难度系数）， S_i 为 N（哪吒）或 W（孙悟空）

接下来 Q 行：第 j 行包含 L_j 和 R_j ，表示第 j 个询问的区间范围

【输出格式】

输出 Q 行。第 j 行输出第 j 个询问的答案（整数）。

【样例 1 输入】

```
1 5 4
2 W 3
3 N 5
4 W 2
5 N 4
6 W 6
7 1 5
8 2 4
9 3 3
10 4 5
```

【样例 1 输出】

```

1 5
2 0
3 0
4 0

```

【样例 1 解释】

$L_1 = 1, R_1 = 5$ 时, 可以将第 5 个位置的标注负责人 W 反转为 N , 则 $[1, 5]$ 样本中, 只有第 1 和第 3 个位置上有 W , 则孙悟空 (W) 负责的样本的标注难度系数之和最小为 $3 + 2 = 5$ 。

$L_2 = 2, R_2 = 4$ 时, 可以将第 3 个位置的标注负责人 W 反转为 N , 则 $[2, 4]$ 样本中, 就没有位置上有 W , 则孙悟空 (W) 负责的样本的标注难度系数之和最小为 0。

$L_3 = 3, R_3 = 3$ 时, 可以将第 3 个位置的标注负责人 W 反转为 N , 则 $[3, 3]$ 样本中, 就没有位置上有 W , 则孙悟空 (W) 负责的样本的标注难度系数之和最小为 0。

$L_4 = 4, R_4 = 5$ 时, 可以将第 5 个位置的标注负责人 W 反转为 N , 则 $[4, 5]$ 样本中, 就没有位置上有 W , 则孙悟空 (W) 负责的样本的标注难度系数之和最小为 0。

【样例 2】

见选手目录下的 *change/change2.in* 与 *change/change2.ans*。

【样例 3】

见选手目录下的 *change/change3.in* 与 *change/change3.ans*。

【数据范围】

- 对于 30% 的数据: $1 \leq n \leq 1000$ 。
- 对于另 40% 的数据: $1 \leq n \leq 10000$ 。
- 对于 100% 的数据: $1 \leq n \leq 10^5$, $1 \leq Q \leq 100$, S_i 为 N 或 W , $1 \leq P_i \leq 10^9$, $1 \leq L_j \leq R_j \leq n$, 且 n, Q, P_i, L_j, R_j 均为整数

神仙跳舞 (dance)

【题目描述】

哪吒和孙悟空各自训练了一个 **AI 舞步生成模型**，用于生成由左右步伐组成的舞蹈编排。哪吒的模型输出由字符串 s 表示，孙悟空的模型输出由字符串 t 表示，其中每个字符为 **0**（左步）或 **1**（右步）。两个模型的输出中均**不包含连续三个相同的步伐**（即没有 **000** 或 **111** 这样的子串）。

在正式演出前的**人工调优阶段**，哪吒可以对模型输出进行最多 K 次**人工修正**，每次修正操作如下：

- 选择 s 或 t 中的一个位置，将该位置的步伐翻转（**0** 变 **1**，**1** 变 **0**）。
- 但修正后，该位置的步伐必须与同一序列中**左右相邻的所有位置**不同。

调优完成后，哪吒将 s 和 t 中的某些步伐配对，形成**"同步点"**。配对必须满足以下所有条件：

- 每个配对由 s 中的一个位置 i 和 t 中的一个位置 j 组成。
- 只有满足 $s_i = t_j$ 的位置才能配对（步伐相同才能同步）。
- s 中的每个位置和 t 中的每个位置**最多只能参与一个配对**。
- 配对之间**不得交叉**。即对于两对 (i, j) 和 (i', j') ，若 $i < i'$ ，则必须满足 $j < j'$ 。

请计算在**最优的人工修正和配对策略**下，最多能形成多少个同步点。

【输入格式】

- 第一行包含字符串 s （由 **0** 和 **1** 组成）。
- 第二行包含字符串 t （由 **0** 和 **1** 组成）。
- 第三行包含整数 K ，表示最多可进行的人工修正次数。

【输出格式】

输出一个整数，表示最多能形成的同步点数量。

【样例 1 输入】

```
1 0101
2 0011
3 1
```

【样例 1 输出】

```
1 3
```


【样例 2 输入】

```
1 01
2 10
3 0
```

【样例 2 输出】

```
1 1
```

【样例 3】

见选手目录下的 *dance/dance3.in* 与 *dance/dance3.ans*。

【样例 4】

见选手目录下的 *dance/dance4.in* 与 *dance/dance4.ans*。

【数据范围】

- 对于 30% 的数据： $1 \leq |s| \leq 10$, $1 \leq |t| \leq 10$
- 对于 100% 的数据： $1 \leq |s| \leq 2 \times 10^3$, $1 \leq |t| \leq 2 \times 10^3$, s 和 t 都仅由 0 和 1 组成, s 和 t 中都不包含三个及以上连续相同字符, $0 \leq K \leq |s| + |t|$, K 为整数, 其中 $|s|$ 表示字符串 s 的字符个数