

NOI 模拟赛 Day2

题目名称	构造题	匹配题	作业题
题目类型	传统型	传统型	传统型
可执行文件名	construct	match	homework
输入文件名	construct.in	match.in	homework.in
输出文件名	construct.out	match.out	homework.out
时间限制	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB
测试点数目	8	20	25
测试点等分	否	是	是

提交源程序文件名

对于 C++ 语言	construct.cpp	match.cpp	homework.cpp
-----------	---------------	-----------	--------------

编译选项

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

1. C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 0。
2. 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格进行分隔。
3. 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。
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 在终端下可使用命令 `ulimit -s unlimited` 将栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
6. 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。
7. 考试过程中若对题目有疑问，请联系出题人。
8. 题目很简单，AK 了请勿大声喧哗。

构造题 (construct)

【题目描述】

给定一个无限大的棋盘，第 i 行第 j 列记作 (i, j) ，棋盘上有若干个位置被标记。你有一棵棋子从 $(1, 1)$ 出发，每次只能向右或向下走一格，经过若干被标记的位置到达 (i, j) ，将这个方案数记作 $f_{i,j}$ ，你想求出所有被标记位置的 $f_{i,j}$ 。

你很轻松地完成了这个题目。毕竟，作为 NOI 选手的你面对这种入门的 dp 题目是毫无压力的。你开始思考这道题目能不能反过来做。给定一个数 k ，由你来选择去标记若干位置，再选择若干被标记的位置，使得它们的 $f_{i,j}$ 之和为 k 。

这个问题仍然是很简单的。但是，你觉得任意一种方案是不够优秀的，你想要被标记的位置和被选择的位置个数都尽量。具体来说，你需要做到被标记的位置数量 $\leq X$ ，被选择的位置数量 $\leq Y$ ， X, Y 为给定的常数，请你构造一组方案来满足条件。如果有多种合法方案，输出任意一种即可。

【输入格式】

第一行四个正整数，分别为 K, Q, X, Y ，保证询问的 $k \leq 10^K$ ， Q 为询问次数。
接下来的 Q 行每行一个正整数 k ，表示询问的 k 。

【输出格式】

第一行一个正整数 n ，表示被标记的位置个数，你需要保证 $n \leq X$ 。

接下来 n 行每行两个数 x, y ，表示被标记的位置坐标。你需要保证被标记的坐标互不相同，并且你需要注意这里的输出顺序（后续需要考虑），尽管棋盘是无限大的，你还是需要保证 $1 \leq x, y \leq n$ 。

接下来 Q 行每行一个长度为 n 的 01 串，其中第 i 位为 1 表示选择了第 i 个输出的位置。你需要保证 1 的个数 $\leq Y$ 。

【样例 1 输入】

```
1 2 3 1000 340
2 1
3 2
4 3
```

【样例 1 输出】

```
1 8
2 1 1
3 1 2
4 1 3
5 2 1
6 2 3
7 3 1
8 3 2
9 3 3
10 10000000
11 00000110
12 10000001
```

【测试方式】

下发文件中提供了 checker.exe，其使用格式为 checker chess.in chess.out，其中 chess.in 为输入文件，chess.out 为你的输出，checker 会执行检查命令并返回错误所对应的编码：

1. 输出格式出错（包括 $n > X$ 但不包括 01 串中 1 的数量 $> Y$ ）。
2. 01 串中 1 的数量 $> Y$ 。
3. 你所构造的方案的 f 值之和不是询问所需要的值。

如果你的构造正确，checker 会返回 correct。

请注意你只能在 windows 系统下使用 checker.exe，否则会出现难以预知的错误。

注意下发 checker 和实际 checker 有所不同。

【子任务】

对于全部的数据， $X \geq 960, Y \geq 240, 1 \leq K \leq 100, 1 \leq Q \leq 10^4$ 。

子任务编号	$X =$	$Y =$	$K =$	特殊性质	分值
1	10^3	340	2	无	5
2	10^3	340	12	无	10
3	10^3	340	100	无	10
4	990	310	100	A	10
5	1050	260	100	无	10
6	1050	240	100	无	10
7	980	260	100	B	15
8	960	240	100	无	30

特殊性质 A: k 在 $[1, 10^K)$ 中均匀随机生成。

特殊性质 B: $Q = 1$ 。

匹配题 (match)

【题目描述】

给定一个长度为 n 的字符串 S 和一个长度为 n 的序列 w_i 。你需要保留 k 个位置，将剩余位置两两匹配起来，形成 $\frac{n-k}{2}$ 个匹配。

对于两个位置 i, j ，它们能够产生匹配当且仅当 $S_i \neq S_j$ ，它们匹配的代价为 $100 \times |i - j| + w_i + w_j$ 。

一个匹配方案的代价为每个匹配的代价之和。求所有匹配方案的代价最小值，若无解输出 -1 。

【输入格式】

本题有多组测试数据。

第一行一个正整数 T ，表示数据组数。

对于每组数据第一行一个字符串 S 。

第二行 n 个正整数表示 $w_1 \sim w_n$ 。

第三行一个正整数 k ，含义见题目描述。

【输出格式】

T 行每行一个数表示每组数据的答案。

【样例 1 输入】

```
1 1
2 aabcde
3 1 1 100000 100000 100000 100000
4 2
```

【样例 1 输出】

```
1 200402
```

【样例 2】

见选手目录下 *match2.in* 与 *match2.ans*，该样例满足测试点 3 ~ 5 的性质。

【样例 3】

见选手目录下 *match3.in* 与 *match3.ans*, 该样例满足测试点 10 ~ 12 的性质。

【样例 4】

见选手目录下 *match4.in* 与 *match4.ans*, 该样例满足测试点 13 ~ 15 的性质。

【子任务】

对于全部的数据, $1 \leq T \leq 5, 1 \leq n \leq 2000, 0 \leq k \leq \min(6, n), 1 \leq w_i \leq 10^5, S_i \in \{a, b, c, d, e, f\}, n \equiv k(\text{mod}2)$ 。

测试点编号	$n \leq$	特殊性质
1 ~ 2	10	无
3 ~ 5	30	无
6 ~ 7	50	无
8 ~ 9	100	无
10 ~ 12	2000	A
13 ~ 15	2000	B
16 ~ 17	2000	C
18 ~ 20	2000	无

特殊性质 A: $S \in \{a, b\}$ 。

特殊性质 B: S_i 在 $\{a, b, c, d, e, f\}$ 中均匀随机生成。

特殊性质 C: $k = 0$ 。

作业题 (homework)

【题目描述】

小 P 有 n 份作业需要做，其中第 i 份作业有 a_i 道题。

他觉得让他自己完成这么多的题目实在是太困难了，所以他找了包括他自己在内的 m 个朋友，每个朋友有自己会的知识点和空闲时间。具体来说，第 i 个朋友可以在 $[l_i, r_i]$ 的作业中挑选至多 c_i 道题目完成。

但是，如果两个朋友不认识，那么他们就很难合作一起帮助小 P 完成作业。我们认为两个朋友认识，当且仅当它们可以完成同样的一份作业。所以小 P 决定选择一份作业，让所有可以完成这份作业的朋友来帮助，他想知道他最多能完成多少道题目。由于小 P 还没有确定选择哪一份作业，所以他想让你对每个 $i \in [1, n]$ 输出让所有能完成作业 i 的朋友帮助的答案。

【输入格式】

第一行两个正整数 n, m ，表示作业数量和朋友数量。

第二行 n 个正整数表示 $a_1 \sim a_n$

接下来 m 行每行三个正整数，表示 l_i, r_i, c_i 。

【输出格式】

一行 n 个正整数，第 i 个表示选择第 i 份作业时的答案。

【样例 1 输入】

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

【样例 1 输出】

```
1 2 5 2 0
```

【样例 2】

见选手目录下 *homework2.in* 与 *homework2.ans*, 该样例满足测试点 5 ~ 7 的性质。

【样例 3】

见选手目录下 *homework3.in* 与 *homework3.ans*, 该样例满足测试点 8 ~ 12 的性质。

【样例 4】

见选手目录下 *homework4.in* 与 *homework4.ans*, 该样例满足测试点 13 ~ 16 的性质。

【子任务】

对于全部的数据, $1 \leq n, m \leq 10^5, 1 \leq l_i \leq r_i \leq n, 1 \leq c_i, a_i \leq 10^9$

测试点编号	$n, m \leq$	特殊性质
1 ~ 2	50	无
3 ~ 4	500	无
5 ~ 7	3000	无
8 ~ 12	10^5	A
13 ~ 16	10^5	B
17 ~ 25	10^5	无

特殊性质 A: $\forall i \in [2, n], l_{i-1} \leq l_i, r_{i-1} \leq r_i$ 。

特殊性质 B: $\forall i, j \in [1, n]$, 保证不存在 $l_i < l_j \leq r_i < r_j$ 。