

NOI 模拟赛

题目名称	跳跃树桩	清除怪兽	括号序列
题目类型	传统型	传统型	传统型
可执行文件名	jump	clear	bracket
输入文件名	jump.in	clear.in	bracket.in
输出文件名	jump.out	clear.out	bracket.out
时间限制	1.0 秒	7.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB
测试点数目	20	25	20
测试点等分	是	是	是

提交源程序文件名

对于 C++ 语言	jump.cpp	clear.cpp	bracket.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。

跳跃树桩 (jump)

【题目描述】

给定 n 个树桩，第 i 个树桩的位置为 x_i ，上面有一个价值为 a_i 的宝物。

小 P 从第一个树桩出发，经过若干树桩跳到第 n 个树桩。他每经过一个树桩，就可以拿走该树桩上的宝物。如果他当前在位置 x ，他可以在一步内跳到 $[x + L, x + R]$ 中的位置。

小 P 对于宝物价值的比对并不是简单地求和，而是有一套特殊的比对方式。具体来说，对于两种方案他分别拿到了 k_1, k_2 个宝物，分别从大到小排序后价值为 $x_1 \sim x_{k_1}$ 和 $y_1 \sim y_{k_2}$ 。那么它会找到满足 $x_i \neq y_i$ 且对于任意 $j < i, x_j = y_j$ 的位置，比较 x_i 和 y_i 的大小关系。如果找不到这种 i 就比较 k_1 和 k_2 的大小关系。通俗来说，将宝物序列按照从大到小排序后字典序比较。

小 P 想求出他能拿到的最优宝物序列，如果不能到达第 n 个树桩，输出 -1 。

【输入格式】

第一行三个正整数 n, L, R 含义见题目描述。

第二行 n 个数表示 $x_1 \sim x_n$ 。

第三行 n 个数表示 $a_1 \sim a_n$ 。

【输出格式】

如果有解第一行输出一个数 k 表示宝物序列长度。

第二行 k 个数表示宝物序列从大到小排序的结果。

如果无解输出 -1 。

【样例 1 输入】

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

【样例 1 输出】

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

【样例 2】

见选手目录下 *jump2.in* 与 *jump2.ans*, 该样例满足测试点 1 ~ 4 的性质。

【样例 3】

见选手目录下 *jump2.in* 与 *jump2.ans*, 该样例满足测试点 5 ~ 7 的性质。

【子任务】

对于全部的数据, 满足 $1 \leq n \leq 5 \times 10^5, 1 \leq x_i, L, R \leq 10^9, 1 \leq a_i \leq n, x_i < x_{i+1}$ 。

测试点编号	$n \leq$	特殊性质
1 ~ 4	500	无
5 ~ 7	5000	有
8 ~ 10	5000	无
11 ~ 14	5×10^5	有
15 ~ 20	5×10^5	无

清除敌人 (clear)

【题目描述】

给定一个 n 个点 m 条边的有向无环图。除了 1 号点以外，其他每个点上都有一个敌人，每个敌人有参数 a_i, b_i ，表示在与它的战斗中需要消耗 a_i 的血量并将其击败，击败后可以获得 b_i 的血量补充。注意，血量低于零意味着死亡，所以需要保证任何时候血量不小于零。

小 Z 从 1 号点出发去击败所有敌人。他每次可以选择沿着图上的边移动一步，或是回到 1 号点。他每次到达一个点的时候，如果这个点的敌人还未被击败，就需要与其进行一场战斗并击败它。问最少需要多少初始血量，他才可以击败所有的敌人。保证 1 号点可以到达所有节点。

【输入格式】

第一行两个数 n, m 表示图的点数和边数。

接下来 $n - 1$ 行每行两个数 a_i, b_i ，第 i 行表示 i 号点上敌人的两个参数。

接下来 m 行每行两个数 u, v ，表示有一条从 u 到 v 的有向边。

【输出格式】

一行一个数，表示答案。

【样例 1 输入】

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

【样例 1 输出】

```
1 4
```

【样例 2】

见选手目录下 *clear2.in* 与 *clear2.ans*，该样例满足测试点 4 ~ 6 的性质。

【子任务】

对于全部的数据，满足 $2 \leq n + m \leq 72, 1 \leq a_i, b_i \leq 10^{15}$ 。

测试点编号	$n \leq$	特殊性质
1 ~ 3	10	无
4 ~ 6	20	无
7 ~ 10	72	$m = n - 1$
11 ~ 16	72	$m \leq n + 10$
17 ~ 21	66	$n + m \leq 66$
22 ~ 25	72	无

括号序列 (bracket)

【题目描述】

合法括号串是满足如下规则的串：

1. 空串是合法括号串。
2. 如果 A 是合法括号串， (A) 也是合法括号串。
3. 如果 A, B 均为合法括号串， AB 也是合法括号串。

给定 n 个盒子，要求你在每个盒子里放入一个合法括号串，需要满足总的左括号数量为 m ，且任意一个盒子中的括号序列长度不能为 $2k$ 。求方案数对质数 mod 取模的结果。

【输入格式】

输入一行四个数， n, m, k, mod ，含义见题目描述。

【输出格式】

一行一个数，表示答案。

【样例 1 输入】

```
1 2 2 1 1000000007
```

【样例 1 输出】

```
1 4
```

【样例 2 输入】

```
1 24 120 30 998244353
```

【样例 2 输出】

```
1 379268651
```

【子任务】

对于全部的数据, $1 \leq n, m \leq 10^7, 1 \leq k \leq m, 10^8 \leq mod \leq 1.01 \times 10^9$ 。保证 mod 为质数。

测试点编号	$n, m, k \leq$	特殊性质
1 ~ 3	500	无
4 ~ 7	8000	$k = m$
8 ~ 10	8000	无
11 ~ 13	10^6	$mod = 998244353$
14 ~ 17	10^7	$k = m$
18 ~ 20	10^7	无