提高级

题目名称	坐标轴（axis）	水符「Bury in Lake」（湖葬）	完美乐队	彩符「彩光乱舞」
题目类型	传统型	传统型	传统型	传统型
目录	axis	dangan	lotus	data
可执行文件名	axis	dangan	lotus	data
输入文件名	axis.in	dangan.in	lotus.in	data.in
输出文件名	axis.out	dangan.out	lotus.out	data.out
每个测试点时限	1000ms	1000ms	5000ms	1000ms
内存限制	512mb	512mb	512mb	512mb
子任务数目	20	4	20	10
测试点是否等分	是	否	是	是

提交源程序文件名

对于C++语言	axis.cpp	dangan.cpp	lotus.cpp	data.cpp

编译选项

对于C++语言	-O2 -lm

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. C/C++中函数main()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. 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）。
4. 程序可使用的栈空间内存限制与题目的内存限制一致。
5. 只提供Linux格式附加样例文件。

坐标轴（axis）(axis)

题目描述

有一个坐标轴，将从 0 到 n 的线段平均分成 n 个区间，第 i 个区间为 $(i-1,i]$。每个区间里有一个物品，第 i 个区间中的物品体积为 a_i，价值为 b_i。

你有 m 行动机会，第 i 次会携带一个容量为 r_i 的包。一次行动从原点出发，向右走过连续的一段后返回，或者在原点直接结束。走过一个区间时，可以选择将当前物品放入包内来获得这个物品，前提是物品体积不大于包的剩余容量。如果没有获得这个物品，则必须将这个物品摧毁。

包的容量单调不减，也就是说有 $r_{i-1} \leq r_i$。

求 m 次行动获得的物品的最大总价值。

输入格式

在文件 `axis.in` 中读入。

第一行两个整数 n, m。

之后的 n 行，每行两个数 a_i, b_i。

之后一行 m 个数表示 r_i。

输出格式

在文件 `axis.out` 中输出。

一个数表示答案。

样例

样例输入 #1

```
5 3
2 10
2 5
1 22
3 7
6 8
1 3 7
```

样例输出 #1

```
44
```

数据范围

对于前 20% 的数据，$n, m \leq 10$。

对于前 40% 的数据，$n, m, a_i, r_i \leq 50$。

另有 20% 的数据，满足 $m \leq 200$。

对于 100% 的数据，$1 \leq n \leq 200$，$1 \leq m \leq 10^5$，$1 \leq a_i \leq 200$，$1 \leq b_i \leq 10^5$，$1 \leq r_i \leq 200$。

水符「Bury in Lake」（湖葬）(dangan)

题目描述

Alex 在和 Ball 下棋，他们面前有一个 $n \times n$ 的空白棋盘 M，Alex 和 Ball 轮流落子，**Ball 先手**。

一个格子最多只能放一枚棋子，同时我们记 Alex 放的为 A 棋子，Ball 放的为 B 棋子。

在某一轮游戏结束后（一轮游戏指的是 Ball 先操作，然后 Alex 操作，总共算一轮），若每一行都有 A 棋子且每一列也都有 A 棋子，则 Alex 立即获胜。

在 $\lfloor n^2/2 \rfloor$ 轮后，若 Alex 仍然没有获胜，则 Ball 获胜。

若 Alex 或 Ball 在某个时刻做了非法的操作（由于遗忘等，在放过棋子的地方重复放），他会直接输掉游戏。

狡猾的 Ball 记忆力好，所以他想了一个办法来恶心 Alex，他要求下盲棋（也就是两人不能随时看棋盘，得用脑子记），且每隔很久很久才给出下一个操作，这样 Alex 落子之前可能会忘掉双方之前的所有操作，只记得 Ball 最近的一步操作。棋盘非常智能，它会在有人满足胜负条件时立刻自动宣布胜负。

现在你得到了这个信息，对于一个给定的 n，你想知道若 Alex 在自己每次操作前忘掉双方之前所有操作，仅知道 Ball 最后一次操作的情况下，是否有必胜策略。

Ball 永远记得双方的所有操作。

由于输入太短，为了避免被 Levin Search 算法解决，因此不提供大样例。

输入格式

在文件 `dangan.in` 中读入。

第一行，一个整数 T，表示数据组数。

接下来 T 行，每行一个整数 n，表示棋盘的大小。

输出格式

在文件 `dangan.out` 中输出。

输出共 T 行，对于每一组数据，输出 `Yes` 或者 `No`。

表示 Alex 在操作前**忘掉**双方之前操作，仅知道 Ball 最后一次操作的情况下他是否一定能获胜。

样例

样例输入 #1

```
2
1
2
```

样例输出 #1

```
No
No
```

数据范围

对于所有数据，$1 \leq n \leq 10^{18}$，$1 \leq T \leq 100$。

详细的数据限制及约定如下（留空表示和上述所有数据的约定相同）：

子任务编号	分值（百分比）	n
1	10	≤ 3
2	40	≤ 100
3	20	$\leq 10^6$
4	30	$\leq 10^{18}$

完美乐队 (lotus)

题目描述

LCR 当上了少女乐队社的指导老师，她要找一群女孩子组乐队。众所周知，一支乐队必须由五个人组成。已经有 $5-k$ 名少女主动报名，现在乐队还差 k（$2 \leq k \leq 5$）个人。

学校里总共还有 N 名少女可以组乐队，第 i 个人擅长的乐器是 a_i，相性是 b_i。

LCR 需要招募 k 名少女，她们擅长的乐器必须互不相同。

并且，少女的相性太接近不利于队内气氛，LCR 不希望这样。也就是说，假设选择的 k 名少女的相性分别是 $b_1, b_2, b_3, \ldots, b_k$，那么，队内气氛的活跃度是 $\min_{i=1}^{k-1} \min_{j=i+1}^{k} |b_i - b_j|$，也就是少女们之间两两相性差（绝对值）的最小值。（原有的主动报名的成员不影响活跃度，只有这 k 个人影响。）

现在，LCR 想组一个队内气氛最活跃的队伍，请你帮他计算出这个最活跃的气氛是多少。

输入格式

在文件 `lotus.in` 中读入。

第一行输入 T，表示一共 T 组测试数据。

对于每组测试数据：

第一行三个正整数 N, k, TP，表示可选的人数，需求的人数，以及测试数据类型，TP 有助于你识别这是哪一类测试数据。

接下来 N 行，每行两个正整数 a_i, b_i，表示第 i 名少女擅长的乐器以及相性值。

输出格式

在文件 `lotus.out` 中输出。

一个数字表示答案（最大可能的队内气氛活跃度），数据保证一定有解。

样例

样例输入 #1

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

样例输出 #1

```
2
```

样例 1 解释

选择第 $\{1,3,5\}$ 名少女，凑齐了三种不同的乐器 $a_1=1,a_3=3,a_5=2$，气氛是 $\min(|1-3|,|3-5|,|1-5|)=2$。

数据范围

测试点 $1\sim2$：$N\leq100$，$TP=1$。

测试点 $3\sim6$：$N\leq1000$，$k=5$，$1\leq a_i\leq5$，$TP=2$。

测试点 $7\sim10$：$N\leq1000$，$k=5$，$TP=3$。

测试点 $11\sim14$：$N\leq10000$，$k=3$，$TP=4$。

测试点 $15\sim17$：$N\leq10000$，$k=5$，$1\leq a_i\leq5$，$TP=5$。

测试点 $18\sim20$：$TP=6$。

对于所有数据：$1\leq N\leq10000$，$2\leq k\leq5$，$1\leq a_i\leq N$，$1\leq b_i\leq10^6$，$T\leq5$。

彩符「彩光乱舞」(data)

题目描述

给定一个长度为 n 的数组 $\{a_n\}$，再给定一个长度为 m 的指令集 t_m。

对于任意一个指令 t_i 有两个参数 l_i, r_i，表示将区间 $[l_i, r_i]$ 中的所有数修改为这个区间的最大值。

有 q 次操作，操作为处理指令集中的指令或者对原数组进行修改，具体的操作有以下两种：

- 给定 L, R, k，求**假设**进行 $t_L, t_{L+1}, \dots, t_R$ 这一段的指令之后，a_k 应该是多少。
- 将 a_u 的值修改为 v。

输入格式

在文件 `data.in` 中读入。

第一行三个整数 n, m, q，表示数组长度，指令集长度和询问数。

第二行 n 个整数 a_i。

接下来 m 行，每行两个整数 l_i, r_i 表示一个指令 t_i。

接下来 q 行，每行表示一次询问，格式如下：

- `1 u v`，则我们将 a_u 改成 v；
- `2 L R k`，请输出**假设**进行 $t_L, t_{L+1}, \dots, t_R$ 这一段的指令之后 a_k 的值。

输出格式

在文件 `data.out` 中输出。

对于每个操作 2 输出一行一个整数表示答案。

样例

样例输入 #1

```
5 3 6
5 3 1 4 2
2 3
3 4
1 5
2 1 2 3
1 4 5
2 1 2 3
1 4 4
1 1 4
2 1 3 1
```

样例输出 #1

```
4
5
4
```

数据范围

对于 30% 的数据，满足 $n, m, q \leq 100$。

对于另外 30% 的数据，满足 $L = 1$。

对于另外 30% 的数据，满足没有操作 1。

对于 100% 的数据，满足 $n, m, q \leq 10^5$，$1 \leq a_i, v \leq 10^5$。