

花生的比赛

题目概况

中文题目名称	天元突破	加拿大	第三题
英文题目与子目录名	struggle	drive	nobb
可执行文件名	struggle	drive	nobb
输入文件名	struggle.in	drive.in	nobb.in
输出文件名	struggle.out	drive.out	nobb.out
每个测试点时限	2 秒	1 秒	1 秒
测试点数目	20	20	10
每个测试点分值	5	5	10
附加样例文件	有	有	有
题目类型	传统	传统	传统
运行内存上限	256M	256M	256M

注意事项:

- 1.结果比较方式 :全文比较（过滤行末空格及文末回车）
- 2.文件名（程序名和输入输出文件名）必须使用英文小写。 3.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
- 4.评测环境听天由命.

天元突破 (struggle.cpp)

题目描述

学习是一个不断克服自己弱点的过程。

花生在经历了许多场爆零的考试之后充分认识到了这一点，同时，他也发现了一个规律：

竞赛考到的知识点一共有 n 个，花生对每个知识点的掌握程度是 A_i 。每次的考试会有 q 道题目，每道题目都有一个唯一的知识点作为考点。考试还有一个难度系数 k ，表示每道题目会从 k 个相邻的知识点中选择一个知识点作为考点。一旦一个知识点考过了，在这场考试中它就不能再出现，并且它将被竞赛之神从知识点中删除。

例如有 5 个知识点，1、2、3、4、5，在考完第 4 个知识点后，剩下的四个知识点就是 1、2、3、5

由于竞赛之神的针对，每一道题目都会从相邻的 k 个知识点中选择一个花生掌握程度最低的知识点来考。

花生不关心考试的成绩，他只想知道他做完这 q 道题的“突破值”是多少。“突破值”是这 q 道题目里他掌握程度最高的知识点与掌握程度最低的知识点 的掌握程度之差。显然这个差值越小，他考试发挥的稳定性就越好。

现在又出现了一场考试，花生只知道考试的题目个数 q 和难度系数 k 。

他想知道，在最好的情况下，他的“突破值”的最小值是多少？

简单来说，你需要自行选定 q 个长度为 k 的区间，每次删除区间内的最小值（当最小值有多个时任选一个），使得删除的 q 个数中最小值和最大值的差尽量小。

输入格式

第一行三个数 n, q, k ，含义如题面所示。

第二行 n 个数，表示 A_1 到 A_n 。

输出格式

一行一个数，突破值的最小值。

样例输入

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

样例输出

```
1
```

样例解释

长度为3的区间有 5、4、1, 4、1、3, 1、3、2 三个,无论选哪一个都必须删除 1, 5、4、3、2中,选择区间4、3、2 并删除2 ,突破值为 1.

数据范围

序号	n取值	k	q	
1	≤ 10	≤ 10	≤ 10	
2	≤ 50		≤ 100	
3	≤ 100			
4	≤ 2000	1		≤ 100
5				
6				
7				
8	≤ 1000	≤ 100	$\leq n$	
9				
10				
11	≤ 1500	$\leq n$		
12	≤ 2000			
13				
14				
15	≤ 100000			$\leq n$
16				
17				
18	≤ 200000	$q \leq 2$		
19		$\leq n$		
20				

加拿大 (drive.cpp)

题目描述

花生和Rockdu要北上加拿大了！他们两个听说加拿大有许多财富，并且免受丧尸的侵扰。

加拿大附近的地形是一张拥有 n 个城市， m 条道路的图，其中有 s 个城市拥有加油站。在出发之前，他们想确定一条去加拿大的路线。现在，他们两个有 Q 种方案前往加拿大，具体来说，每种方案都有一个起点城市 S 和终点城市 T ，并且在每种方案中，他们会选择一辆车，这种车每次加满油之后可以连续行驶 K 公里。为了保证他们能顺利到达加拿大，他们选择的起点城市和终点城市都拥有加油站。而每当他们经过拥有加油站的城市时，他们都会停下来把油加满。

Rockdu和花生对于到底选择哪一种方案争执不下。于是他们决定先排除一些不能到达加拿大的路线——一旦在旅途中油用尽了，而他们没有到达任何一个拥有加油站的城市，他们就没办法去往加拿大了。Rockdu 和花生都不在乎他们走的路径是否最短，也不在乎经过了多少个城市，他们只要你判断对于每种方案，他们能找到一条路径到达终点吗？

输入格式

第一行三个整数 n, m, s ，其中 $s \leq n$ 。

接下来一行 s 个数，表示拥有加油站的城市编号，

接下来 m 行每行有三个数 u_i, v_i, d_i ，表示一条连接 u_i, v_i ，长度为 d_i 的路径。

接下来一行一个整数 Q 。

接下来 Q 行，每行有三个数 S_i, T_i, K_i ，含义如上所示。

输出格式

共 Q 行，对于每个方案，如果能到达加拿大，则输出 *TAK*，否则输出 *NIE*。

样例输入

```
6 5 4
1 5 2 6
1 3 1
2 3 2
3 4 3
4 5 5
6 4 5
4
1 2 4
2 6 9
1 5 9
6 5 8
```

样例输出

```
TAK
TAK
TAK
NIE
```

数据范围

序号	n	m	s	q	
1	10	20	<=n	10	
2	20	30			
3	30	40			
4	40	50			
5	5000	n-1		15	100000
6	60000				
7	200000	200000	100		
8					
9					
10					
11	150000	200000	<=n	200000	
12					
13					
14					
15					
16	200000				
17					
18					
19					
20					

提示

这张图不一定联通.

第三题 (nobb.cpp)

题目描述

花生在出完题之后已经很累了，并且他的加拿大之旅即将开始。他不想再说话，所以题目描述也变得很简单。

给定一个数列 a_n , m 个函数 f_n , 每个函数都是数列的一个区间之和. 即 $f_i = \sum_{l_i}^{r_i} a_i$

要求支持两个操作:

- 1、单点修改 a_p 的值.
- 2、询问 f_l 到 f_r 所有函数的和.

输入格式

第一行两个数 n, m ，分别表示数列的长度和函数的个数。

第二行 n 个数，表示 a_1, a_2, \dots, a_n

接下来 m 行，每行 2 个数 l_i, r_i ，表示第 i 个函数所对应的区间。

接下来一行一个数 k ，表示有 k 个操作。

接下来 k 行，每行 3 个数 $type, u, v$ 。

$type = 1$ 时，该操作是 1 操作，将 a_u 的值修改为 v 。

$type = 2$ 时，该操作是 2 操作，询问 f_u 到 f_v 的和。

输入保证合法，读入的所有数均为小于 10^9 的自然数。

输出格式

对于每个 2 操作，输出一行一个整数，表示给定函数的和。

样例输入

5 5
2 4 2 7 3
1 1
1 5
3 4
2 5
2 5
4
2 4 5
2 3 5
1 1 5
2 1 2

样例输出

4
14
29
32

数据范围

序号	n	k	m
1	100	5000	1000
2			
3	5000	100	
4			
5	5000	5000	20000
6			
7			
8			
9	20000	20000	
10			