

准考证号：SH-B260105

(请选手务必仔细阅读本页内容)

一、题目概况

中文题目名称	星际探险	删除子数组	轮流值班	堵车	黄浦江
英文题目与子目录名	explore	subarray	duty	congestion	river
可执行文件名	explore	subarray	duty	congestion	river
输入文件名	explore.in	subarray.in	duty.in	congestion.in	river.in
输出文件名	explore.out	subarray.out	duty.out	congestion.out	river.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒	1 秒
测试点数目	10	10	10	10	10
每个测试点分值	10	10	10	10	10
附加样例文件	无	无	无	无	无
结果比较方式	全文比较，过滤末行后空行，不过滤中间行行末空格				
题目类型	传统	传统	传统	传统	传统
运行内存上限	512M	512M	512M	512M	512M

二、提交源程序文件名

对于 C++ 语言	explore.cpp	subarray .cpp	duty .cpp	congestion.cpp	river.cpp
-----------	-------------	---------------	-----------	----------------	-----------

三、编译命令（不包含任何优化开关）

对于 C++ 语言	g++ -o explore -1m	g++ -o subarray subarray -1m	g++ -o duty duty.cpp -1m	g++ -o congestion.cpp congestion.cpp -1m	g++ -o river river .cpp -1m
-----------	--------------------	------------------------------	--------------------------	--	-----------------------------

星球探索

(explore.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

经过了长达 2 年的星际飞行，普罗米修斯号飞船终于进入了 LV-223 号星球的轨道。这是人类第一次进行如此距离的星际探险活动，探险队的任务是来这里寻找人类的起源。刚下飞船，探险队就遇到了一个难题，这颗星球的气候十分恶劣，山峰之间的高度，随便一座都能完爆地球上的喜马拉雅山，同时飞船扫描显示这颗星球的大气结构不适合人类呼吸，探险一度陷入了僵局。幸运的是，探险队有地质勘探球可以使用，这些小球可以沿着星球地面飞行并进行扫描，将信息传回飞船，在飞船上建立起全息影像，。

小球现在从地点 0 依次经过地点 1，地点 2直到地点 N ，共 $N + 1$ 个地点。最终目的地在地点 N 。地点 0 的海拔 $A_0 = 0$ ，地点 i 的海拔为 A_i 。

小球的电池量随海拔升降而变化。地点 0 在海边，电池量 0；对于任一地点 i ，从地点 i 前往地点 $i + 1$ 的电池消耗量仅取决于两地的海拔差。

如果 $A_i = A_{i+1}$ ，小球的电量不变；

如果 $A_i < A_{i+1}$ ，小球每爬升 1 米，电量就会下降 S ；

如果 $A_i > A_{i+1}$ ，小球每下沉 1 米，电量就会升高 T 。

LV-223 号星球的地壳运动很强烈。你得到了 Q 天来地壳运动的数据。在第 j 日，地点 $L_j, L_j + 1, L_j + 2, \dots, R_j$ 的海拔升高了 X_j ，注意 X_j 可能是负数。

你的任务是，计算每天地壳运动后 小球探索后的电池量（可以是负数）。

【输入格式】

输出文件 explore.in

第一行有四个整数 N, Q, S, T ，用空格分隔。

在接下来的 $N + 1$ 行中，第 i 行 ($1 \leq i \leq N + 1$) 有一个整数 A_{i-1} 。

在接下来的 Q 行中，第 j 行 ($1 \leq j \leq Q$) 有三个整数 L_j, R_j, X_j ，用空格分隔。

输入的所有数的含义见题目描述。

【输出格式】

输出文件 explore.out

共 Q 行，第 j 行 ($1 \leq j \leq Q$) 有一个整数，表示第 j 日地壳运动后小球探索的最终电量。

【输入输出样例 1】

explore.in	explore.out
3 5 1 2	-5
0	-7
4	-13
1	-13
8	-18
1 2 2	
1 1 -2	
2 3 5	
1 2 -1	
1 3 5	

【说明】

地点 0,1,2,3 的海拔分别是 0,4,1,8。第一天地壳运动后，海拔分别为 0,6,3,8。此时电池电量分别为 0,-6,0,-5。

【输入输出样例 2】

explore.in	explore.out
2 2 5 5	5
0	-35
6	
-1	
1 1 4	
1 2 8	

【输入输出样例 3】

explore.in	explore.out
7 8 8 13	277
0	277
4	322
-9	290
4	290

-2	290
3	290
10	370
-9	
1 4 8	
3 5 -2	
3 3 9	
1 7 4	
3 5 -1	
5 6 3	
4 4 9	
6 7 -10	

【数据规模与约定】

对于 30%的数据： $N, Q \leq 2000$

对于另外 10%的数据： $S = T$

对于 100%的数据： $1 \leq N, Q \leq 2 \times 10^5, 1 \leq S, T \leq 10^6, A_0 = 0$

$|A_i| \leq 10^6 (1 \leq i \leq N), |X_j| \leq 10^6, 1 \leq L_j \leq R_j \leq N (1 \leq j \leq Q)$

删除子数组

(subarray.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

在数组中截取一组连续的元素可以构成子数组。例如数组 {1, 7, 2, 3, 4, 5} 的子数组可以是 {1}, {2, 3, 4}, {2, 3, 4, 5} 等等, 但 {1, 2, 3} 不是子数组。现在给定正整数数组 A 和某个正整数 x, 可以删除其中的一个子数组, 要求删除后剩余的元素总和可以被 x 整除。但是不允许将数组中所有元素都删除掉。

显然满足要求的方案可能不止一种, 要求在所有的合法删除方案中, 找出最短的被删除子数组, 并输出该子数组的长度。如果不存在合法的删除方案, 输出 -1。

【输入格式】

输入文件 subarray.in

输入包括 N+1 行:

第一行是一个整数 N 和 x, N 表示数组的长度, 删除子数组后剩余元素和可以被 x 整除。

接下来 N 行的每一行一个数字, 表示数组的元素。

【输出格式】

输出文件 subarray.out

输出满足要求的最短子数组的长度, 找不到合法的就输出 -1

【输入输出样例 1】

subarray.in	subarray.out
4 6 3 1 4 2	1

【说明】

数组 中元素和为 10, 不能被 x 整除。我们可以移除子数组 {4}, 剩余元素的和为 6。

【输入输出样例 2】

subarray.in	subarray.out
3 7 1 2 3	-1

【说明】没有任何方案使得移除子数组后剩余元素的和被 7 整除。

【输入输出样例 3】

subarray.in	subarray.out
3 3 1 2 3	0

【说明】和恰好为 6，已经能被 3 整除了。所以我们不需要移除任何元素。

【数据规模与约定】

$1 < A[i] \leq 10^9, 1 \leq x \leq 10^9$

20%数据， $N \leq 100$

50%数据， $N \leq 1000$

100%数据， $N \leq 100000$

轮流值班

(duty.cpp)

时空限制：1s/512M，测试数据共 10 组

【问题描述】

你是保安大队队长，负责门房间的值班工作。目前共有 n 名队员参与值班，编号 1 到 n 。其中， i 号队员可以覆盖的工作时间段为时刻 a_i 到时刻 b_i ，覆盖的时间长度为 $(b_i - a_i)$ 。

根据这些信息，你可以推算出能够覆盖的时间段总长度。

但是，有一件棘手的事情，你需要带领一名队员去其他城市出差，参加全国安保大会。为了使剩下的 $n-1$ 名队员能够覆盖的时间长度尽量长，你会仔细挑选出差的人选。请问剩下的 $n-1$ 名队员能够覆盖的时间长度最长是多少？

注意：如果部分时间段由多名队员覆盖，那么也只会算作一次，不会重复计算。

【输入格式】

输入文件 duty.in

输入的第一行是一个正整数 n 。后面的 n 行每行两个整数 a_i, b_i 。

【输出格式】 输出文件 duty.out

输出共 1 个整数。

【输入输出样例 1】

duty.in	duty.out
3	7
5 9	
1 4	
3 7	

【说明】 选第 3 名队员出差，另外两人覆盖了 $(9-5)+(4-1)=4+3=7$ 的时间长度。

【输入输出样例 2】

duty.in	duty.out
3	8
1 9	
1 4	
1 3	

【数据规模与约定】

测试数据共 10 组

1 号数据: $n=3$

2 号数据: $n=5$

3 号数据: $n \leq 10, 0 \leq a_i \leq b_i \leq 1000$

4 号数据: $n \leq 100, 0 \leq a_i \leq b_i \leq 1000$

所有数据: $2 \leq n \leq 100000, 0 \leq a_i \leq b_i \leq 1000000000$

葉奧编程

堵车

(congestion.cpp)

时空限制：1s/512M，测试数据共 10 组

【题目描述】

在繁华的市中心，交通网络错综复杂，分布着 n 个关键的路口，它们通过 m 条双向道路相互连接，每条道路都对应着特定的行驶时间。对于居住在市中心的小戈来说，每天的通勤路线是从起点 1 号路口出发，直达终点 n 号路口。在这个交通网络中，任意两个路口之间最多只有一条直接相连的道路，不存在冗余的多条道路连接同一对路口。无论何时，小戈总能通过一系列道路找到一条从起点到终点的可行路径。在规划行程时，小戈总是力求选择一条能够使总行驶时间最短的路线。

不幸的是，有一天市中心发生了一起严重的交通事故，导致某条道路的行驶时间翻了一倍。由于事故的具体位置未知，小戈无法确定这次事故是否会影响到他的通勤路线。在最理想的情况下，事故可能与小戈的路线完全无关，他的通勤时间不会受到任何影响。然而，在最糟糕的情况下，事故可能会增加他的通勤时间。通过对市中心交通网络的深入分析，我们需要计算出在最不利的情况下，小戈从 1 号路口到 n 号路口的行驶时间可能会增加多少。

【输入格式】

从文件 congestion.in 中读入数据。

输入第一行为正整数 n 和 m ，用空格分隔。

后续 m 行中，每行包含三个正整数 A_i , B_i , C_i ，代表第 i 条道路信息。其中 A_i 和 B_i 是在 1 到 n 内的路口编号，表示由道路连接的两个路口， C_i 是道路的行驶时间（不超过 10^6 ）。

【输出格式】

输出到文件 congestion.out 中。

输出共一行，包含一个整数，表示最差情况下，最短路径行驶时间的变化量。

【输入输出样例 1】

congestion.in	congestion.out
5 7	2
2 1 5	
1 3 1	
3 2 8	
3 5 7	

3 4 3	
2 4 7	
4 5 2	

【说明】

有 5 个路口和 7 条道路。没有事故时，从 1 号路口到 n 号路口的最短行驶路径是 1-3-4-5，总行驶时间为 $1+3+2=6$ 。如果在 3 号路口到 4 号路口的道路出现事故，其行驶时间从 3 增加到 6，那么最短行驶路径现在是 1-3-5，总行驶时间为 $1+7=8$ ，比之前的最短行驶路径时间增加了 2。

【样例 2】

见目录下的 congestion2.in 与 congestion2.ans。

【数据规模与约定】

测试点编号	特殊性质
1~2	$m=n-1$, $A_i=i$, $B_i=i+1$, 符合链条形态
3~4	$n \leq 10$
5~6	$n \leq 100$
7~10	无

保证所有数据： $n \leq 500$, $m \leq 100000$ 。

黄浦江

(river.cpp)

时空限制：1s/512M，测试数据共 10 组

【题目描述】

魔都要进行超级城市的建设，计划在黄浦江底下铺设更多的江底隧道。黄浦江两边一边是浦东一边是浦西，每一边都有 n 个可能的隧道出入口，从南到北依次编号都是 1 到 n 。你作为魔都市长，收到 m 条隧道的设计方案，你都非常喜欢，但是为了安全起见，隧道不能出现在江底交叉的情况，但可以共享出入口。请问你至少要否决掉几条隧道方案？

【输入格式】

输入文件为 river.in

输入一行为正整数 n 和 m ， $2 \leq n \leq 10000$ ， $m \leq 200000$ 。接着输入 m 行，每行有 2 个正整数 x_i, y_i ，均在 1 到 n 之间。**【输出格式】**

输出文件为 river.out 输出一个整数。

【输入输出样例 1】

river.in	river.out
2 3 1 2 1 2 2 1	1

【说明】 去掉 2-1 这个隧道即可。**【输入输出样例 2】**

river.in	river.out
4 5 4 3 3 4 4 1 2 1 1 2	2

【说明】 去掉 1-2 和 3-4 这两条隧道即可。

【数据规模与约定】

20%的数据: $n \leq 10, m \leq 20$

50%的数据: $n \leq 2000, m \leq 5000$

100%的数据: $n \leq 10000, m \leq 200000$

葉奧编程