

NOIP 模拟赛

题目名称	AddEdge Loves Working	LargeDumpling Loves Graph	jvjhfhg Loves Sequence
可执行文件名	work	graph	sequence
输入文件名	work.in	graph.in	sequence.in
输出文件名	work.out	graph.out	sequence.out
每个测试点时限	1 秒	2 秒	2 秒
内存限制	64MB	256MB	256MB
测试点数目	20	20	20
每个测试点分值	5	5	5
是否有部分分	否	否	否
结果比较方式	全文比较（忽略行末空格和文末回车）		
题目类型	传统型	传统型	传统型
是否有附加文件	否	否	否

提交源程序需加后缀

对于 C++ 语言	work.cpp	graph.cpp	sequence.cpp
-----------	----------	-----------	--------------

编译开关

对于 C++ 语言	-lm	-lm	-lm
-----------	-----	-----	-----

AddEdge Loves Working

(work.cpp)

【问题描述】

AddEdge 在 Tenhou 雀馆工作。
Tenhou 雀馆非常高级,所有自动麻将桌的电源由中央电脑直接控制。Tenhou 雀馆还非常大,你可以认为有无数个包间。AddEdge 的工作是操控中央电脑开关机麻电源并指示前台将新来的客人带到哪个包间。但是他觉得每次都要开关电源太麻烦了,于是他决定如果从上一批客人离开到下一批客人到来不超过 m 单位时间,他就不关电源。他已经拿到了这一天的预约清单,请帮助他计算他最多可以少关多少次电源。

【输入格式】

输入文件名为 *work.in*。
第一行两个整数 n 和 m , 表示预约的客人批数和最长的不关电源的时间
接下来 n 行, 每行两个整数 s_i 与 t_i , 表示第 i 批客人预约到来和预约使用包间的时间。

【输出格式】

输出文件名为 *work.out*。
输出仅包含一个整数表示答案。

【样例输入与输出】

work.in	work.out
3 5 1 5 6 3 14 6	2
5 10 2 6 1 2 17 7 3 9 15 6	3

【数据范围】

对于 50%的数据: $n \leq 1000$
对于 100%的数据: $n \leq 300000, 1 \leq m, s_i, t_i \leq 10^9$
数据有梯度。

LargeDumpling Loves Graph

(graph.cpp)

【问题描述】

LargeDumpling 最近参加了志愿家教活动，但是第二次小朋友又鸽了，于是他在纸上玩起了无向图。

LargeDumpling 在纸上画了一张 n 个点 m 条边的无向图，他希望知道一些点对之间的最短路。但是他觉得这样不够刺激，于是他想知道它们之间恰好经过 e 条边的最短路。

LargeDumpling 在 1s 内就在脑中算出了答案，但是他觉得现在的计算机都太菜了，所以你的程序在 2s 内算出答案即可。

【输入格式】

输入文件名为 **graph.in**。

第一行为三个整数 n, m, e 。

接下来 m 行，每行三个整数 u, v, w ，表示 u 到 v 有一条长度为 w 的边。

接下来一行一个整数 q ，表示询问组数。

接下来 q 行每行两个整数 u, v ，表示询问 u 到 v 的最短路。

数据保证图无重边无自环，保证不会询问一对相同的点。

【输出格式】

输出文件名为 **graph.out**。

对于每组询问输出一行一个整数表示答案，若题述路径不存在，输出-1。

【样例输入与输出】

graph.in	graph.out
4 4 2	2
1 2 1	-1
1 3 1	3
2 4 2	
4 3 2	
3	
2 3	
1 2	
1 4	

【数据范围】

对于 20% 的数据： $n \leq 5, q \leq 10$

对于 50% 的数据： $n \leq 50, 1 \leq e \leq 100, q \leq 50$

对于 70% 的数据： $n \leq 50, 1 \leq e \leq 100$

对于 100% 的数据： $n \leq 100, 1 \leq e \leq m \leq \frac{n \times (n-1)}{2}, q \leq 10000, 0 < w \leq 1000$

jvjhfhg Loves Sequence

(sequence.cpp)

【问题描述】

圣诞节到了, jvjhfhg 的儿子们收到了来自父亲的礼物以及贺卡。jvjhfhg 在贺卡上出了一道题给儿子们:

给出一个长度为 n 的序列, 给定常数 m 和 k , 询问 q 次, 每次询问区间 $[L, R]$ 中有多少个子区间满足区间中第 k 大的数不小于 m 。

【输入格式】

输入文件名为 *sequence.in*。

第一行三个正整数 n, m, k 。

第二行 n 个正整数 a_i , 表示序列。

第三行一个正整数 q 。

接下来 q 行, 每行两个正整数 L, R , 保证 $L \leq R$ 。

【输出格式】

输出文件名为 *sequence.out*。

对于每个询问输出一行表示答案。

【样例输入与输出】

sequence.in	sequence.out
7 4 2	18
4 2 7 7 6 5 1	5
3	13
1 7	
2 5	
1 6	

【数据范围】

对于 20% 的数据: $n, q \leq 50$

对于 50% 的数据: $n, q \leq 50000, k \leq 10$

对于另外 20% 的数据: $q = 1$ 且保证 $L = 1, R = n$

对于 100% 的数据: $n, q \leq 200000, k \leq n \leq 200000, 1 \leq a_i, m \leq 10^9$

保证仅对于第三类数据有 $q = 1$ 。