|  |  |  |  |
| --- | --- | --- | --- |
| 题目名称 | 奇怪的城市 | 最小的序列 | 港记的速度 |
| 目录 | walk | sequence | journalist |
| 可执行文件名 | walk | sequence | journalist |
| 输入文件名 | walk.in | sequence.in | journalist .in |
| 输出文件名 | walk.out | sequence.out | journalist .out |
| 每个测试点时限 | 1s | 1s | 1s |
| 内存限制 | 128MB | 128MB | 128MB |
| 测试点数量 | 10 | 10 | 10 |
| 每个测试点分值 | 10 | 10 | 10 |
| 是否有部分分 | 否 | 否 | 否 |
| 题目类型 | 传统 | 传统 | 传统 |

**NOIP水题赛Day1**

*~~这里真的是个蒟蒻求轻喷QAQ~~*

By jzq

提交源程序后缀：

|  |  |  |  |
| --- | --- | --- | --- |
| 对于C++语言 | walk.cpp | sequence.cpp | journalist.cpp |
| 对于C语言 | walk.c | sequence.c | journalist.c |
| 对于Pascal语言 | walk.pas | sequence.pas | journalist.pas |

**注意本次评测将会开启O2优化开关**

**奇怪的城市(walk.cpp/c/pas)**

**【题目背景】**

前几日@787赋闲在家，他闲着无聊于是买了张机票飞到了Wa城，顺便呢也去机场找好货（~~可惜一个都没捕捉到23333~~

**【题目描述】**

Wa城中有一条笔直的左右无限长的道路，道路上有n座住宅，每座住宅居住着一位居民。同时这条道路上的某一个位置是市中心。

现在我们为这n座住宅标号，第i座住宅的坐标X[i]表示这座住宅相对于市中心的位置。如果X[i]>0，则住宅在市中心右边X[i]个单位的位置，如果X[i]<0，则住宅在市中心左边，没有住宅在市中心这个位置，每个X[i]保证是偶数，且没有两座住宅在同一位置。

@787对这里进行观察之后发现，这些居民有一些奇怪的习惯。每天他们都会同时出门，出门后都会只往一个方向行走，每单位时间行走一单位距离，并且不会改变方向。而且每当他们相互碰见，他们就会停下来，和对方像Wallace一样谈笑风生，直到这一天结束。

现在给出n座住宅的位置，和某一天他们出门的行走方向，求在第T个单位时间的时候第P位居民的位置（假设这一天有无限个单位时间）

**【输入描述】**

文件第一行输入两个整数n、q。q表示询问的个数

第2~n+1行，每行两个整数X[i]、F[i]。X[i]意义同上，F[i]表示这位居民这一天行走方向，1为向右走，2为向左走

接下来q行，每行两个整数T、P。意义同上

**【输出描述】**

q行，每行一个整数，表示这位居民当前的坐标。

**【样例输入】**

6 3

-10 1

-6 2

-4 1

2 1

6 2

18 2

1 2

6 4

7 6

**【样例输出】**

-7

4

11

**【数据规模及约定】**

对于20%数据，n<=100，T<=10000

对于40%数据，n<=5000

另外有20%数据，从最左边数起连续的若干居民全部往右，剩下的全部往左

对于100%数据，|X[i]|<=10^15，T<=10^15,1<=q<=n<=100000

保证读入时X[i]升序，而且不会有两位居民初始位置重合。

**最小的序列(sequence.cpp/c/pas)**

**【题目背景】**

在飞往Wa城的途中，@787不时地向空乘搭讪，以至于他居然得到了在落地之后进cockpit参观的机会（我也想要>\_<），参观结束后机长送给他一样小礼物，这是一个类似栈的容器（你可以理解为就是一个栈），容器内最多可以放置m个数。

**【题目描述】**

最近@787突发奇想要用这个容器构造序列。一开始他有一个n个正整数的初始序列，我们需要将其中的数按顺序压入这个容器并弹出。这些数按它们的弹出顺序进行排列，会得到一个新的序列。当然啦，这样可以构造出很多不同的序列，但是@787要求这个序列字典序最小（字典序最小意为若第一个数最小的序列有多个，则令第二个数最小。若仍有多个，则第三个最小，以此类推）。请你帮他构造出来吧。

**【输入描述】**

第一行输入两个整数n、m，接下来一行n个正整数表示初始序列

**【输出描述】**

一行n个正整数表示经过构造后字典序最小的序列

**【样例输入】**

6 3

5 2 3 8 7 4

**【样例输出】**

2 3 5 4 7 8

**【数据规模及约定】**

对于10%数据，n<=10

对于50%数据，n<=5000

对于另10%数据，保证随机

对于100%数据，n<=300000，每个数不超过10^9

**港记的速度(journalist.cpp/c/pas)**

**【题目背景】**

@787回来后突然对香港记者那无与伦比的速度产生了兴趣(蛤)。所以对这帮特殊的群体展开了研究。

**【题目描述】**

@787发现香港记者平时的速度都是练出来的。他们是通过一个n\*m的地图来训练的，地图的某些位置是墙（柱子），人不能通过而且也不透光，另外的地方是一些空地（当然透光）。空地的某一个位置是一位需要拍摄/采访的对象（~~长者~~）。记者能向四个方向行走（上，下，左，右），每走一步（移动一格）用时1s。记者们还能向八个方向看（上，右上，右，右下，下，左下，左，左上）。我们假设记者们的视力非常好，他能够看到非常非常远的地方直到这条光线被墙/柱子之类的东西挡住。 记者们的任务就是从一个初始位置（当然是空地）出发，到达一个能够看到需要拍摄/采访的对象（~~长者~~）的位置，当然啦用时肯定要最小（这样就跑得快了啊）。

**【输入描述】**

第一行输入两个整数n、m，接下来是一个n\*m的地图，“@”表示这个位置是一堵墙/柱子，“O”表示这个位置是空地。

接下来有若干行，每行四个正整数sx,sy,ex,ey表示初始位置和拍摄/采访的对象的位置，以“0 0 0 0”结束

**【输出描述】**

每一行对于每一种位置的情况输出记者们看到对象的最短时间，如果不能看到的话，输出“Naive!”

**【样例输入】**

3 4

O@@O

@@OO

@OOO

3 2 2 4

3 3 1 1

0 0 0 0

**【样例输出】**

1

Naive!

**【数据规模及约定】**

对于30%数据，n，m<=10

对于100%数据，n，m<=200，询问数<=200

看到这的人相信你们都AK了吧。。。