**写在前面的话：**

这绝对是最简单的一次测试，也应该是你从未参加过的测试类型。但也绝对是对大家知识的吸收和遗忘程度一种最好检验的测试。

这是一套完完整整的最近一次做的练习题！

本次测试限时3个小时，即11:30结束。

**求逆序对数目**

(deseq)

**题目描述**

给定一个序列a1,a2,…,an，如果存在i<j并且ai>aj，那么我们称之为逆序对的，求逆序对的数目。

**输入**

第一行为n,表示序列长度，接下来的n行，第i+1行表示序列中的第i个数。

**输出**

所有逆序对总数。

**样例输入**

4

3

2

3

2

**样例输出**

3

**提示**

N<=105，Ai<=105

**末日传说**

(legend))

**题目描述**

在古老的东方，人们都采用一种奇妙的方式记录日期：他们用一些特殊的符号来表示从1开始的连续整数，1表示最小而N表示最大。创世纪的第一天，日历就被赋予了生命，它自动的开始计数，就像排列在不断地增加。

我们用1——N来表示日历的元素，第一天的日历是：1,2,3,……N-1,N

第二天，日历自动变为1,2,3,……N,N-1

每次他都生成一个以前从未出现过的“最小”的排列——将它转为（N+1）进制后数的数值最小。

有一天，一个预言者出现了——虽然没人让他出现——他预言道，当这个日历到达某个上帝安排的时刻（……），这个世界就会崩溃……他还预言，假如一个某一个日子的逆序到达一个指定数值M时，世界末日将来临。

逆序是什么？日历中两个不同的符号，假如排在前面的那个比后面的那个大，就是一个逆序（不一定相邻）。人们期待一个贤者来预见那一天。

**输入描述** Input Description

只包含一行两个非负整数，分别为N和M。

**输出描述** Output Description

输出一行，为世界末日的日期，每个数字间用一个空格隔开。

**样例输入** Sample Input

5 4

**样例输出** Sample Output

1 3 5 4 2

**数据范围及提示** Data Size & Hint

对于10%的数据有N<=10。

对于40%的数据有N<=1000。

对于100%的数据有N<=50000。M<=1000000000

所有数据均有解。

**聚会的快乐**

**(****party)**

**描述**

你要组织一个由你公司的人参加的聚会。你希望聚会非常愉快，尽可能多地找些有趣的热闹。但是劝你不要同时邀请某个人和他的上司，因为这可能带来争吵。给定N个人（姓名，他幽默的系数，以及他上司的名字），找到能使幽默系数和最大的若干个人。

**输入**

第一行一个整数N（N<100）。接下来有N行，每一行描述一个人，信息之间用空格隔开。姓名是长度不超过20的字符串。幽默系数是在0到100之间的整数。如果一个人没有上司，那么他的上司会写成NOBODY

**输出**

邀请的人最大的幽默系数和。

**样例输入**

5

BART 1 HOMER

HOMER 2 MONTGOMERY

MONTGOMERY 1 NOBODY

LISA 3 HOMER

SMITHERS 4 MONTGOMERY

**样例输出**

8

**好文章**

**(B.cpp)**

**【问题描述】**

nodgd 写了一篇文章，自认为这是一篇好文章。nodgd 的文章由n个小写英文字母组成。文章的一个子串指的是文章中的一段连续的字母，子串的长度就是这一段的字母个数。nodgd 在文章中用了排比、对偶、前后照应之类的手法，所以就有很多个子串是相同或者相近的。为了向大家证明这是一篇好文章，nodgd决定给自己的文章进行评分。nodgd 首先确定了一个整数m，然后统计出文章中有多少个不相同的长度为m的子串，这个数量就是文章的评分。然而，nodgd 懒得老老实实计算这个评分了，就把任务丢给了你。

**【输入格式】**

输入文件 B.in。

第一两个整数n,m，表示文章的长度和需要统计的子串长度。

第二行包含一个长度为n的只包含小写字母的字符串。

**【输出格式】**

文件 B.out。

输出一行一个整数，表示文章的评分。

**【样例输入 1】**

5 3

aaaab

**【样例输出 1】**

2

**【样例解释 1】**

长度为3的子串有3个，分别是 aaa,aaa,aab，其中不同的只有2个。

**【样例输入 2】**

9 3

abcabacba

**【样例输出 2】**

7

**【样例解释 2】**

共有7个长度为3的子串，每个长度为3的子串都不同。

**【数据范围】**

于 30%的数据，1 ≤m≤ n≤ 200；

于 50%的数据，1 ≤m≤ n≤ 2000；

于另外 20%的数据，1 ≤ m ≤ 50 ≤ n ≤ 200000

于另外 100%的数据，1 ≤ m ≤ n ≤ 200000

**有点不同**

**（****little）**

**题目描述：**

最小生成树是图论中一个很常见的问题。对你来说应该也是很简单的。现在这道题和普通的最小生成树有点不同。

给定一个带权无向图G(V, E)，如果T是G的一棵生成树，定义

value(T) = max{ value(e) | e is in T } – min{ value(e) e is in T}，即value(T)是这棵生成树中最大权值的边与最小权值的边之差。

现在，我要你找出最小的value(T)。

**输入格式(little.in):**

第一行是两个整数N和M，分别表示图G的顶点数和边数。

然后是M行，第i (1<=i<=M) 行含三个整数Xi (0<Xi<=N), Yi (0<Yi<=N), Di(0<Di<=10^8), 表示图G中有一条边(Xi, Yi)，边权为Di。

输入保证G是连通的，而且没有平行边（即两个点之间最多只有一条边）。

**输出格式(little.out):**

输出一个整数，表示最小的value(T)。

**输入样例：**

3 3

1 2 10

1 3 20

2 3 30

**输出样例：**

10

**数据说明：**

对于50%的数据，1<=N<=20，M <= 200。

对于100%的数据，1<=N<=100，M <= 5000。

**Black box**

**（****bbox）**

**描述**

黑盒子代表一个原始的数据库，它可以保存一个整数数组，并且有一个特殊变量i。在开始时候，黑盒子是空的，i=0。黑盒子处理一系列的命令，这里有两种命令：

ADD(x)：将x放入到黑盒子中；

GET：i增加1，然后给出黑盒子中第i小的数（第i小的数的定义：黑盒子中所有元素按照非降序排列，排在第i位的元素）

看一个11个序列的处理方式：

i 空盒子中的数 GET的值

1 ADD(3) 0 3   
2 GET 1 3 3   
3 ADD(1) 1 1, 3   
4 GET 2 1, 3 3   
5 ADD(-4) 2 -4, 1, 3   
6 ADD(2) 2 -4, 1, 2, 3   
7 ADD(8) 2 -4, 1, 2, 3, 8   
8 ADD(-1000) 2 -1000, -4, 1, 2, 3, 8   
9 GET 3 -1000, -4, 1, 2, 3, 8 1   
10 GET 4 -1000, -4, 1, 2, 3, 8 2   
11 ADD(2) 4 -1000, -4, 1, 2, 2, 3, 8

你需要编写程序完成这些操作，ADD和GET操作每种不超过30000个

我们用两个整数数组描述命令序列

1. A(1), A(2), ..., A(M) 应该要包括的元素序列，保证每个数的绝对值都不超过2 000 000 000，M<=30000,上述例子中的A={3, 1, -4, 2, 8, -1000, 2}
2. u(1), u(2), ..., u(N)一个序列，表示在ADD第几个元素的时候进行GET操作，例子中u=（1, 2, 6, 6）

我们可以假设u中的序列是非下降子序列，N<=M

**输入描述：**

第一行：M N

第二行M个数，表示A

第三行N个数，表示u

**输出描述**

Get到的数值，每行一个

**Sample Input**

7 4

3 1 -4 2 8 -1000 2

1 2 6 6

**Sample Output**

3

3

1

2

**排列布局**

（layout）

**描述**

当排队等候喂食时，奶牛喜欢和它们的朋友站得靠近些。FJ有N（2<=N<=1000）头奶牛，编号从1到N，沿一条直线站着等候喂食。奶牛排在队伍中的顺序和它们的编号是相同的。因为奶牛相当苗条，所以可能有两头或者更多奶牛站在同一位置上。即使说，如果我们想象奶牛是站在一条数轴上的话，允许有两头或更多奶牛拥有相同的横坐标。一些奶牛相互间存有好感，它们希望两者之间的距离不超过一个给定的数L。另一方面，一些奶牛相互间非常反感，它们希望两者间的距离不小于一个给定的数D。给出ML条关于两头奶牛间有好感的描述，再给出MD条关于两头奶牛间存有反感的描述。（1<=ML,MD<=10000，1<=L,D<=1000000）你的工作是：如果不存在满足要求的方案，输出-1；如果1号奶牛和N号奶牛间的距离可以任意大，输出-2；否则，计算出在满足所有要求的情况下，1号奶牛和N号奶牛间可能的最大距离。

**输入：**

第一行：N,ML,MD

接下来ML行，每行三个整数：a,b,d,表示牛a和牛b的距离最多为d.

接下来MD行，每行三个整数：a,b,d,表示牛a和牛b的距离最多为d.

**输出：**

如果不存在满足要求的方案，输出-1；如果1号奶牛和N号奶牛间的距离可以任意大，输出-2；否则，计算出在满足所有要求的情况下，1号奶牛和N号奶牛间可能的最大距离。

**输入样例：**

4 2 1

1 3 10

2 4 20

2 3 3

**输出样例：**

27

**样例说明：**

四只牛分别在0,7,10,27