

一、陶陶摘苹果

(apple.pas/c/cpp)

【问题描述】

陶陶家的院子里有一棵苹果树，每到秋天树上就会结出 10 个苹果。苹果成熟的时候，陶陶就会跑去摘苹果。陶陶有个 30 厘米高的板凳，当她不能直接用手摘到苹果的时候，就会踩到板凳上再试试。

现在已知 10 个苹果到地面的高度，以及陶陶把手伸直的时候能够达到的最大高度，请帮陶陶算一下她能够摘到的苹果的数目。假设她碰到苹果，苹果就会掉下来。

【输入文件】

输入文件 **apple.in** 包括两行数据。第一行包含 10 个 100 到 200 之间（包括 100 和 200）的整数（以厘米为单位）分别表示 10 个苹果到地面的高度，两个相邻的整数之间用一个空格隔开。第二行只包括一个 100 到 120 之间（包含 100 和 120）的整数（以厘米为单位），表示陶陶把手伸直的时候能够达到的最大高度。

【输出文件】

输出文件 **apple.out** 包括一行，这一行只包含一个整数，表示陶陶能够摘到的苹果的数目。

【样例输入】

```
100 200 150 140 129 134 167 198 200 111
110
```

【样例输出】

```
5
```

二、校门外的树

(tree.pas/c/cpp)

【问题描述】

某校大门外长度为 L 的马路上有一排树，每两棵相邻的树之间的间隔都是 1 米。我们可以把马路看成一个数轴，马路的一端在数轴 0 的位置，另一端在 L 的位置；数轴上的每个整数点，即 0, 1, 2, …, L ，都种有一棵树。

由于马路上有一些区域要用来建地铁。这些区域用它们在数轴上的起始点和终止点表示。已知任一区域的起始点和终止点的坐标都是整数，区域之间可能有重合的部分。现在要把这些区域中的树（包括区域端点处的两棵树）移走。你的任务是计算将这些树都移走后，马路上还有多少棵树。

【输入文件】

输入文件 **tree.in** 的第一行有两个整数 L ($1 \leq L \leq 10000$) 和 M ($1 \leq M \leq 100$)， L 代表马路的长度， M 代表区域的数目， L 和 M 之间用一个空格隔开。接下来的 M 行每行包含两个不同的整数，用一个空格隔开，表示一个区域的起始点和终止点的坐标。

【输出文件】

输出文件 **tree.out** 包括一行，这一行只包含一个整数，表示马路上剩余的树的数目。

【样例输入】

```
500 3
150 300
100 200
470 471
```

【样例输出】

```
298
```

【数据规模】

对于 20%的数据，区域之间没有重合的部分；
对于其它的数据，区域之间有重合的情况。

三、采药

(**medic.pas/c/cpp**)

【问题描述】

辰辰是个天资聪颖的孩子，他的梦想是成为世界上最伟大的医师。为此，他想拜附近最有威望的医师为师。医师为了判断他的资质，给他出了一个难题。医师把他带到一个到处都是草药的山洞里对他说：“孩子，这个山洞里有一些不同的草药，采每一株都需要一些时间，每一株也有它自身的价值。我会给你一段时间，在这段时间里，你可以采到一些草药。如果你是一个聪明的孩子，你应该可以让采到的草药的总价值最大。”

如果你是辰辰，你能完成这个任务吗？

【输入文件】

输入文件 **medic.in** 的第一行有两个整数 T ($1 \leq T \leq 1000$) 和 M ($1 \leq M \leq 100$)，用一个空格隔开， T 代表总共能够用来采药的时间， M 代表山洞里的草药的数目。接下来的 M 行每行包括两个在 1 到 100 之间（包括 1 和 100）的整数，分别表示采摘某株草药的时间和这株草药的价值。

【输出文件】

输出文件 **medic.out** 包括一行，这一行只包含一个整数，表示在规定的时间内，可以采到的草药的最大总价值。

【样例输入】

```
70 3
71 100
69 1
1 2
```

【样例输出】

```
3
```

【数据规模】

对于 30% 的数据， $M \leq 10$ ；

对于全部的数据， $M \leq 100$ 。

四、循环

(circle.pas/c/cpp)

【问题描述】

乐乐是一个聪明而又勤奋好学的孩子。他总喜欢探求事物的规律。一天，他突然对数的正整数次幂产生了兴趣。

众所周知，2 的正整数次幂最后一位数总是不断的在重复 2，4，8，6，2，4，8，6……我们说 2 的正整数次幂最后一位的循环长度是 4（实际上 4 的倍数都可以说是循环长度，但我们只考虑最小的循环长度）。类似的，其余的数字的正整数次幂最后一位数也有类似的循环现象：

| | 循环 | 循环长度 |
|---|---------|------|
| 2 | 2、4、8、6 | 4 |
| 3 | 3、9、7、1 | 4 |
| 4 | 4、6 | 2 |
| 5 | 5 | 1 |
| 6 | 6 | 1 |
| 7 | 7、9、3、1 | 4 |
| 8 | 8、4、2、6 | 4 |
| 9 | 9、1 | 2 |

这时乐乐的问题就出来了：是不是只有最后一位才有这样的循环呢？对于一个整数 n 的正整数次幂来说，它的后 k 位是否会发生循环？如果循环的话，循环长度是多少呢？

注意：

1. 如果 n 的某个正整数次幂的位数不足 k ，那么不足的高位看做是 0。
2. 如果循环长度是 L ，那么说明对于任意的正整数 a ， n 的 a 次幂和 $a + L$ 次幂的最后 k 位都相同。

【输入文件】

输入文件 **circle.in** 只有一行，包含两个整数 n ($1 \leq n < 10^{100}$) 和 k ($1 \leq k \leq 100$)， n 和 k 之间用一个空格隔开，表示要求 n 的正整数次幂的最后 k 位的循环长度。

【输出文件】

输出文件 **circle.out** 包括一行，这一行只包含一个整数，表示循环长度。
如果循环不存在，输出 -1。

【样例输入】

32 2

【样例输出】

4

【数据规模】

对于 30%的数据， $k \leq 4$ ；

对于全部的数据， $k \leq 100$ 。