



<h1>A</h1>	<h1>Cyclic Triangle</h1>
	<div>Time Limit</div> <div>5 seconds</div>

กำหนดพื้นที่สามเหลี่ยมมุมฉากขนาด $N \times N$ เราต้องการเขียนตัวเลขเป็นรูปสามเหลี่ยม ในลักษณะเป็นวงวนพร้อมนับค่าไปเรื่อย ๆ โดยจะเริ่มเขียนในแนวนอนในบรรทัดก่อน จากนั้นจึงต่อลงมาในแนวตั้ง และวกกลับขึ้นไปในแนวทแยง ทั้งนี้การนับจะนับจาก 1 ถึง 9 แล้วเริ่มต้นใหม่ที่ 0 และนับต่อไปที่ 1 2 3 ... เป็นเช่นนี้ไปเรื่อย ๆ เช่น ถ้า $N = 10$ เราจะได้หมายเลขในบริเวณพื้นที่สามเหลี่ยมเป็นดังแสดงข้างล่าง

```

1 2 3 4 5 6 7 8 9 0
  7 8 9 0 1 2 3 4 1
    6 5 6 7 8 9 5 2
      5 4 4 5 0 6 3
        4 3 3 1 7 4
          3 2 2 8 5
            2 1 9 6
              1 0 7
                0 8
                  9

```

อย่างไรก็ตาม สิ่งที่เราสนใจมากกว่าก็คือว่า ถ้าหากทราบค่า N แล้ว เราจะสามารถหาค่าตัวเลขที่ตำแหน่งแถวและคอลัมน์ที่ต้องการได้อย่างรวดเร็วอย่างไร เช่น จาก $N = 10$ ถ้าตำแหน่งคือ แถว 0 คอลัมน์ 7 คำตอบคือ 8 แต่ถ้าเป็น แถว 5 คอลัมน์ 7 คำตอบคือ 2 เป็นต้น **จงเขียนโปรแกรมที่คำนวณเลขในตำแหน่งที่สนใจได้อย่างรวดเร็ว**

ข้อมูลเข้า

บรรทัดแรกคือจำนวนเต็มบวก $T \leq 10$ ซึ่งแสดงจำนวนชุดทดสอบ และแต่ละชุดทดสอบข้อมูลเข้าเป็นดังนี้

บรรทัดแรกคือค่า N และ Q โดย N แทนขนาดของสามเหลี่ยมและ Q แทนจำนวนตำแหน่งที่สนใจ ซึ่ง $5 \leq N \leq 1,000,000$ และ $2 \leq Q \leq 10$ อีก Q บรรทัดต่อมาเป็นตำแหน่งที่สนใจ หนึ่งตำแหน่งต่อบรรทัด ในรูปแบบ $R\ C$ เมื่อ R คือแถว และ C คือคอลัมน์ โดยที่ $0 \leq R \leq C < N$ (ในที่นี้แถวและคอลัมน์นับจาก 0)

ผลลัพธ์

เป็นตัวเลขที่ตำแหน่งที่สนใจ หนึ่งตัวเลขต่อบรรทัด

หมายเหตุ ในผลลัพธ์จะมีบรรทัดเปล่าตามท้ายตัวเลขตัวสุดท้ายมาด้วย

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
2	2
12 5	1
7 8	3
0 0	6
11 11	1
5 10	8
2 7	0
1000000 6	0
7000 8000	4
5212 6824	8
700000 999999	3
5000 72913	
900000 955555	
8775 57412	

ตัวอย่าง

ตารางข้างล่างนี้ แสดงภาพสามเหลี่ยมเมื่อ $N = 11$, 12 และ 13 เราสามารถใช้ภาพนี้ในการทดสอบโปรแกรมได้

11	12 (เลขที่เน้นคือตำแหน่งที่ถูกอ้างอิงถึงในตัวอย่าง)
<pre> 1 2 3 4 5 6 7 8 9 0 1 0 1 2 3 4 5 6 7 8 2 9 1 2 3 4 5 6 9 3 8 0 3 4 5 7 0 4 7 9 2 6 8 1 5 6 8 1 9 2 6 5 7 0 3 7 4 6 4 8 3 5 9 2 0 1 </pre>	<pre> 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 2 7 8 9 0 1 2 3 3 4 1 6 2 3 4 5 4 4 5 0 5 1 8 6 5 5 6 9 4 0 7 6 6 7 8 3 9 7 7 8 7 2 8 8 9 6 1 9 0 5 0 1 4 2 3 </pre>

13	
1 2 3 4 5 6 7 8 9 0 1 2 3 6 7 8 9 0 1 2 3 4 5 6 4 5 3 4 5 6 7 8 9 0 7 5 4 2 1 2 3 4 5 1 8 6 3 1 0 0 1 6 2 9 7 2 0 9 9 7 3 0 8 1 9 8 8 4 1 9 0 8 7 5 2 0 9 7 6 3 1 8 6 4 2 7 5 3 6 4 5	