



พักร้อน (Holiday)

เจียนเจียกำลังวางแผนเที่ยวพักร้อนครั้งต่อไปที่ไต้หวัน ระหว่างช่วงเวลาที่เขาพักร้อนอยู่ เจียนเจียจะเดินทางจากเมืองหนึ่งไปยังอีกเมืองหนึ่ง และเยี่ยมชมแหล่งท่องเที่ยวต่าง ๆ ในเมืองเหล่านั้น

ในไต้หวันมีเมืองอยู่ n เมือง ทุกเมืองตั้งอยู่บนถนนสายเดียวกัน เมืองเหล่านี้จะมีตัวเลขกำกับเรียงไว้ตั้งแต่ 0 ถึง $n - 1$ เมืองที่อยู่ติดกับเมือง i เมื่อ $0 < i < n - 1$ คือเมือง $i - 1$ และเมือง $i + 1$ เมืองที่อยู่ติดกับเมือง 0 คือเมือง 1 เท่านั้น ส่วนเมือง $n - 1$ อยู่ติดกับเมือง $n - 2$ เท่านั้น

เมืองแต่ละเมืองจะมีจำนวนแหล่งท่องเที่ยวแตกต่างกัน เจียนเจียมีเวลาพักร้อน d วัน เขาวางแผนที่จะไปเยี่ยมชมแหล่งท่องเที่ยวให้ได้มากที่สุด โดยเขาได้เลือกเมืองที่จะเริ่มต้นในการพักร้อนของเขาไว้แล้ว ในแต่ละวันเจียนเจียสามารถเลือกได้ว่าเขาจะเดินทางจากเมืองที่เขาอยู่ไปยังเมืองที่อยู่ติดกัน หรือจะไปเยี่ยมชมแหล่งท่องเที่ยวทุกแห่งของเมืองที่เขาอยู่ในวันนั้น แต่เขาไม่สามารถที่จะทำทั้งสองอย่างพร้อมกันได้ในวันเดียว เจียนเจียจะไม่ยอมไปเยี่ยมชมแหล่งท่องเที่ยวซ้ำในเมืองเดิมโดยเด็ดขาดถึงแม้ว่าเขาจะสามารถมาพักอยู่ที่เมืองเดิมได้หลายครั้ง กรุณาช่วยเจียนเจียวางแผนเที่ยวพักร้อนเพื่อให้เขาไปเยี่ยมชมแหล่งท่องเที่ยวได้มากที่สุด

ตัวอย่าง

สมมุติว่าเจียนเจียมีวันหยุดพักร้อนทั้งหมด 7 วัน มีเมืองทั้งหมด 5 เมือง (ตามตารางด้านล่าง) เขาเริ่มต้นที่เมือง 2 โดยในวันแรกเจียนเจียเลือกที่จะเยี่ยมชมแหล่งท่องเที่ยวทั้ง 20 แห่งในเมือง 2 วันที่สองเจียนเจียเดินทางจากเมือง 2 ไปเมือง 3 และในวันที่สามเขาเยี่ยมชมแหล่งท่องเที่ยวทั้ง 30 แห่งในเมือง 3 จากนั้นเจียนเจียใช้เวลาอีกสามวันเพื่อเดินทางจากเมือง 3 ไปยังเมือง 0 และในวันที่เจ็ดเขาเยี่ยมชมแหล่งท่องเที่ยวทั้ง 10 แห่งในเมือง 0 ดังนั้นจำนวนแหล่งท่องเที่ยวทั้งหมดที่เจียนเจียได้ไปเยี่ยมชมคือ $20 + 30 + 10 = 60$ ซึ่งถือเป็นจำนวนแหล่งท่องเที่ยวที่มากที่สุดที่เจียนเจียสามารถไปเยี่ยมชมได้ภายในเวลา 7 วันเมื่อเขาเริ่มต้นจากเมือง 2

| เมือง | จำนวนแหล่งท่องเที่ยว |
|-------|----------------------|
| 0 | 10 |
| 1 | 2 |
| 2 | 20 |
| 3 | 30 |
| 4 | 1 |

| วันที่ | กิจกรรม |
|--------|----------------------------------|
| 1 | เยี่ยมชมแหล่งท่องเที่ยวในเมือง 2 |
| 2 | เดินทางจากเมือง 2 ไปเมือง 3 |
| 3 | เยี่ยมชมแหล่งท่องเที่ยวในเมือง 3 |
| 4 | เดินทางจากเมือง 3 ไปเมือง 2 |
| 5 | เดินทางจากเมือง 2 ไปเมือง 1 |
| 6 | เดินทางจากเมือง 1 ไปเมือง 0 |
| 7 | เยี่ยมชมแหล่งท่องเที่ยวในเมือง 0 |

งานของคุณ

คุณต้องเขียนฟังก์ชัน `findMaxAttraction` เพื่อคำนวณหาจำนวนแหล่งท่องเที่ยวที่มากที่สุดที่เจี้ยนเจียจะไปเยี่ยมชมได้

- `findMaxAttraction(n, start, d, attraction)`
 - `n`: จำนวนเมืองทั้งหมด
 - `start`: หมายเลขของเมืองเริ่มต้น
 - `d`: จำนวนวันทั้งหมด
 - `attraction`: อาร์เรย์ขนาด n โดยที่ `attraction[i]` คือจำนวนแหล่งท่องเที่ยวในเมือง i โดยที่ $0 \leq i \leq n - 1$
 - ฟังก์ชันนี้ต้องคืนค่าเป็นจำนวนเมืองที่มากที่สุดที่เจี้ยนเจียสามารถไปเยี่ยมชมได้

งานย่อย

สำหรับงานย่อยทั้งหมด เรารับประกันว่า $0 \leq d \leq 2n + \lfloor n/2 \rfloor$ และจำนวนแหล่งท่องเที่ยวในแต่ละเมืองมีค่าไม่เป็นลบ

ข้อจำกัดเพิ่มเติม

| งานย่อย | คะแนน | n | จำนวนแหล่งท่องเที่ยวสูงสุด ในหนึ่งเมือง | เมืองเริ่มต้น |
|---------|-------|-------------------------|--|---------------|
| 1 | 7 | $2 \leq n \leq 20$ | 1,000,000,000 | เมืองใดก็ได้ |
| 2 | 23 | $2 \leq n \leq 100,000$ | 100 | เมือง 0 |
| 3 | 17 | $2 \leq n \leq 3,000$ | 1,000,000,000 | เมืองใดก็ได้ |
| 4 | 53 | $2 \leq n \leq 100,000$ | 1,000,000,000 | เมืองใดก็ได้ |

รายละเอียดการเขียนโปรแกรม

คุณจะต้องส่งแฟ้มเพียงแฟ้มเดียวเท่านั้น โดยมีชื่อเป็น `holiday.c`, `holiday.cpp` หรือ `holiday.pas` โดยภายในแฟ้มนี้จะต้องมีโปรแกรมน้อยที่ทำงานตามคำอธิบายข้างต้นโดยใช้รูปแบบด้านล่าง ถ้าคุณใช้ภาษา C/C++ คุณจะต้อง include header `holiday.h`

กรุณาสั่งเกตว่าคำตอบที่เป็นไปได้ อาจจะมีขนาดใหญ่มาก และชนิดของตัวแปรที่คืนมาจาก `findMaxAttraction` นั้นเป็นจำนวนเต็ม 64 บิต

โปรแกรม C/C++

```
long long int findMaxAttraction(int n, int start, int d,
int attraction[]);
```

โปรแกรมปาสคาล

```
function findMaxAttraction(n, start, d : longint;
attraction : array of longint): int64;
```

เกรตเตอร์ตัวอย่าง

เกรตเตอร์ตัวอย่างจะอ่านค่าอินพุตในรูปแบบต่อไปนี้

- บรรทัดที่ 1: $n, start, d$
- บรรทัดที่ 2: $attraction[0], \dots, attraction[n-1]$

เกรตเตอร์ตัวอย่างจะพิมพ์ผลลัพธ์ของค่าที่คืนออกมาจาก `findMaxAttraction`