



## 假期 (Holiday)

健佳正在制定下个假期去台湾的游玩计划。在这个假期，健佳将会在城市之间奔波，并且参观这些城市的景点。

在台湾共有  $n$  个城市，它们全部位于一条高速公路上。这些城市连续地编号为  $0$  到  $n-1$ 。对于城市  $i$  ( $0 < i < n-1$ ) 而言，与其相邻的城市是  $i-1$  和  $i+1$ 。但是对于城市  $0$ ，唯一与其相邻的是城市  $1$ 。而对于城市  $n-1$ ，唯一与其相邻的是城市  $n-2$ 。

每个城市都有若干景点。健佳有  $d$  天假期并且打算要参观尽量多的景点。健佳已经选择了假期开始要到访的第一个城市。在假期的每一天，健佳可以选择去一个相邻的城市，或者参观所在城市的所有景点，但是不能同时进行。即使健佳在同一个城市停留多次，他也不会去重复参观该城市的景点。请帮助健佳策划这个假期，以便能让他参观尽可能多的景点。

### 例子

假设健佳有 7 天假期，有 5 个城市（参见下表），而且他由城市 2 开始。在第一天，健佳参观城市 2 的 20 个景点。第二天，健佳由城市 2 去往城市 3。而在第三天，健佳参观城市 3 的 30 个景点。接下来的 3 天，健佳由城市 3 前往城市 0。而在第 7 天，健佳参观城市 0 的 10 个景点。这样健佳参观的景点总数是  $20+30+10=60$ ，这是他由城市 2 开始、在 7 天假期内最多能参观的景点数目。

城市	景点数目
0	10
1	2
2	20
3	30
4	1

第几天	活动
1	参观城市 2 的景点
2	由城市 2 前往城市 3
3	参观城市 3 的景点
4	由城市 3 前往城市 2
5	由城市 2 前往城市 1
6	由城市 1 前往城市 0
7	参观城市 0 的景点

## 任务

请实现函数findMaxAttraction，以计算健佳最多可以参观多少个景点。

- findMaxAttraction(*n*, start, *d*, attraction)
  - *n*: 城市数。
  - start: 起点城市的编号。
  - *d*: 假期的天数。
  - attraction: 长度为 *n* 的数组；attraction[*i*] 表示城市 *i* 的景点数目，其中  $0 \leq i \leq n - 1$ 。
  - 该函数应返回健佳最多可以参观的景点数。

## 子任务

在所有的子任务中，有  $0 \leq d \leq 2n + \lfloor n/2 \rfloor$ 。而且，每个城市中的景点数都是非负整数。

附加限制：

子任务	分值	<i>n</i>	各城市景点数的最大值	起点城市
1	7	$2 \leq n \leq 20$	1,000,000,000	无限制
2	23	$2 \leq n \leq 100,000$	100	城市 0
3	17	$2 \leq n \leq 3,000$	1,000,000,000	无限制
4	53	$2 \leq n \leq 100,000$	1,000,000,000	无限制

## 实现细节

你只能提交一个文件，文件名为 holiday.c、holiday.cpp 或 holiday.pas。该文件中需要实现如上所述的函数，并遵循如下命名与接口信息。如果你编写的是 C/C++ 程序，则还需要加入头文件 holiday.h。

注意，计算的结果可能会很大，所以函数 findMaxAttraction 的返回值类型是一个 64 位整数。

### C/C++ 程序

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

### Pascal 程序

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

### 评测方式

评测系统需要读入如下格式的输入数据：

- 第1行:  $n, \text{start}, d$ .
- 第2行:  $\text{attraction}[0], \dots, \text{attraction}[n-1]$ .

评测系统将会输出 `findMaxAttraction` 的返回值。