



Արձակուրդ

Ձիան-Ձիան պլանավորում է իր հաջորդ արձակուրդն անցկացնել Թայվանում: Արձակուրդի ընթացքում Ձիան-Ձիան քաղաքից քաղաք է անցնում և այցելում այդ քաղաքների ատրակցիոնները:

Թայվանում կա n քաղաք, դրանք բոլորը տեղակայված են միակ մայրուղու երկայնքով: Քաղաքները համարակալված են հաջորդաբար 0-ից $n - 1$ թվերով: i -րդ քաղաքի, որտեղ $0 < i < n - 1$, հարակից քաղաքներն են $i - 1$ -րդը և $i + 1$ -րդը: 0 քաղաքի միակ հարակից քաղաքը 1-ն է, $n - 1$ -րդ քաղաքի միակ հարակից քաղաքը $n - 2$ -ն է:

Յուրաքանչյուր քաղաքում կան ինչ-որ թվով արտարկցիոններ: Ձիան-Ձիան պլանավորում է այցելել որքան հնարավոր է շատ ատրակցիոններ: Ձիան-Ձիան արդեն ընտրել է, թե որ քաղաքում պետք է սկսի իր արձակուրդը: Արձակուրդի յուրաքանչյուր օր Ձիան-Ձիան կարող է կամ տեղափոխվել հարակից քաղաք, կամ այցելել այն քաղաքի բոլոր ատրակցիոնները, որտեղ նա գտնվում է, բայց ոչ երկու գործողությունը նույն օրում: Ձիան-Ձիան *որևէ քաղաքում նույն ատրակցիոնը երկու անգամ չի այցելում* անգամ եթե նա այդ քաղաքում մի քանի անգամ է մնում: Օգնեք, խնդրեմ, Ձիան-Ձիային պլանավորել արձակուրդն այնպես, որ նա այցելի որքան հնարավոր է շատ արտրակցիոններ:

Օրինակ

Դիցուք Ձիան-Ձիան ունի 6 օր արձակուրդ, կա 5 քաղաք (թվարկված ներքևի աղյուսակում), և նա սկսում է քաղաք 2-ից: Առաջին օրը Ձիան-Ձիան այցելում է քաղաք 2-ի 20 ատրակցիոնները: Երկրորդ օրը Ձիան-Ձիան քաղաք 2-ից տեղափոխվում է քաղաք 3: Երրորդ օրը նա այցելում է քաղաք 3-ի 30 ատրակցիոնները: Հաջորդ երկու օրերին Ձիան-Ձիան քաղաք 3-ից տեղափոխվում է քաղաք 1 և վեցերորդ օրը այցելում է քաղաք 1-ի 10 ատրակցիոնները: Ձիան-Ձիայի այցելած ատրակցիոնների ընդհանուր քանակը կլինի $20 + 30 + 10 = 60$, որը ատրակցիոնների մեծագույն քանակն է, որ Ձիան-Ձիան կարող է այցելել 6 օրում, եթե սկսի քաղաք 2-ից:

քաղաք	ատրակցիոնների քանակը
0	2
1	10
2	20
3	30
4	1

օր	գործողություն
-----------	----------------------

օր	գործողություն
1	այցելել քաղաք 2-ի ատրակցիոնները
2	քաղաք 2-ից տեղափոխվել քաղաք 3
3	այցելել քաղաք 3-ի ատրակցիոնները
4	քաղաք 3-ից տեղափոխվել քաղաք 2
5	քաղաք 2-ից տեղափոխվել քաղաք 1
6	այցելել քաղաք 1-ի ատրակցիոնները

Խնդիր

Իրականացրեք, խնդրեմ, `findMaxAttraction` ֆունկցիան, որը հաշվում է, թե Չիան-Չիան առավելագույնը քանի ատրակցիոն կարող է այցելել:

- `findMaxAttraction(n, start, d, attraction)`
 - n -ը քաղաքների քանակն է:
 - `start`-ը սկզբնական քաղաքի համարն է:
 - d -ն օրերի քանակն է:
 - `attraction`-ը n երկարության զանգված է: `attraction[i]`-ն ցույց է տալիս i -րդ քաղաքի ատրակցիոնների քանակը, որտեղ $0 \leq i \leq n - 1$.
 - Ֆունկցիան պետք է վերադարձնի, թե Չիան-Չիան առավելագույնը քանի ատրակցիոն կարող է այցելել:

Ենթախնդիրներ

Բոլոր ենթախնդիրներում $0 \leq d \leq 2n + \lfloor n/2 \rfloor$.

Լրացուցիչ սահմանափակումներ.

ենթախնդիր	միավոր	n	քաղաքում ատրակցիոնների մաքսիմալ քանակը	սկզբնակս քաղաքը
1	7	$2 \leq n \leq 20$	$0 \leq t \leq 1,000,000,000$	սահմանափակում չկա
2	23	$2 \leq n \leq 100,000$	$0 \leq t \leq 100$	0 քաղաք
3	17	$2 \leq n \leq 3,000$	$0 \leq t \leq 1,000,000,000$	սահմանափակում չկա
4	53	$2 \leq n \leq 100,000$	$0 \leq t \leq 1,000,000,000$	սահմանափակում չկա

Իրականացման մանրամասներ

Պահանջվում է հանձնել ճիշտ մեկ ֆայլ, որի անունը պետք է լինի `holiday.c`, `holiday.cpp` կամ `holiday.pas`: Այդ ֆայլում պետք է իրականացնել վերը նկարագրված ֆունկցիաները օգտագործելով հետևյալ նախատիպերը: C/C++ ծրագրերում պետք նաև ավելացնել `holiday.h` ֆայլի ընդգրկման հրամանը:

Նկատենք, որ արդյունքը կարող է մեծ լինել, և որ `findMaxAttraction`-ի վերադարձի տիպը 64-բիթանոց ամբողջ է:

C/C++ ծրագիր

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

Pascal ծրագիր

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

Գրելիների օրինակ

Գրելիների օրինակը մուտքային տվյալները ներածում է հետևյալ ձևաչափով.

- Տող 1. `n`, `start`, `d`:
- Տող 2. `attraction[0]`, ..., `attraction[n-1]`:

Գրելիների օրինակը կարտածի `findMaxAttraction` ֆունկցիայի վերադարձի արժեքը: