



Vakáció

Jian-Jia vakációján tajvani városokat látogat meg. Az n város egy út mentén helyezkedik el, a városokat 0 -tól $n - 1$ -ig sorszámozzuk. Az i . szomszédja ($0 < i < n - 1$) az $i - 1$. és az $i + 1$. város. A 0 -nak csak az 1 ., az $n - 1$ -nek csak az $n - 2$. a szomszédja.

Minden városban van valahány látványosság, Jian-Jia úgy akar városokat meglátogatni, hogy a lehető legtöbb látványosságot nézze meg. Eldöntötte, hogy melyik városban kezdi a szabadságát. Minden nap két lehetőség közül választhat: megnézi az adott városban az összes látványosságot vagy átmegy egy szomszédos városba. (Amit egyszer megnézett, azt többször már nem nézheti meg.)

Segíts Jian-Jia-nak kiszámítani a megnézendő látványosságok maximális számát!

Példa

Jian-Jia 7 napot nyaral, 5 város van (a táblázatbeli látványosságokkal), a 2. városból indul. Az első napon megnéz 20 látványosságot a 2. városban. A második napon átmegy a 3. városba, a harmadik napon megnéz 30 látványosságot a 3. városban. A következő 3 napon átmegy a 3. városból az 0.-ba, majd a hetedik napon megnézi az ott levő 10 látványosságot. Tehát a látványosságok maximális száma, amit meg tud nézni 7 nap alatt a 2. városból indulva: $20 + 30 + 10 = 60$.

város	látványosságok száma
0	10
1	2
2	20
3	30
4	1

nap	tevékenység
1	megnézi a 2. város látványosságait
2	átmegy a 2. városból a 3.-ba
3	megnézi a 3. város látványosságait
4	átmegy a 3. városból a 2.-ba
5	átmegy a 2. városból az 1.-be
6	átmegy az 1. városból a 0.-ba
7	megnézi a 0. város látványosságait

Feladat

A `findMaxAttraction` függvényt kell megírnod, amely kiszámítja a megnézendő látványosságok

maximális számát!

- `findMaxAttraction(n, start, d, attraction)`
 - `n`: a városok száma.
 - `start`: a kezdő város sorszáma.
 - `d`: a napok száma.
 - `attraction`: n elemű tömb; `attraction[i]` az i . városban levő látványosságok száma ($0 \leq i \leq n - 1$).
 - a megnézendő látványosságok maximális számát kell visszatéríteni!

Részfeladatok

Minden tesztesetben $0 \leq d \leq 2n + \lfloor n/2 \rfloor$.

Korlátok:

részfeladat	pont	n	látványosságok száma (t)	kezdő város
1	7	$2 \leq n \leq 20$	$0 \leq t \leq 1,000,000,000$	nincs korlát
2	23	$2 \leq n \leq 100,000$	$0 \leq t \leq 100$	0. város
3	17	$2 \leq n \leq 3,000$	$0 \leq t \leq 1,000,000,000$	nincs korlát
4	53	$2 \leq n \leq 100,000$	$0 \leq t \leq 1,000,000,000$	nincs korlát

Megvalósítás

A `holiday.c`, `holiday.cpp` vagy `holiday.pas` fájlt kell beküldened! Ebben kell megvalósítanod a megoldó függvényt! Include-old a `holiday.h`-t!

A `findMaxAttraction` eredménye nagyon nagy lehet, használj 64-bites egész számot!

C/C++ program

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

Pascal program

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

Minta értékelő

A minta értékelő a bemenetet az alábbi formában várja:

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

A minta értékelő kiírja a `findMaxAttraction` értékét.