



დღესასწაული

ჯიან-ჯი მომავალი სადღესასწაულო დღეების გატარებას ტაივანში გეგმავს. მას უნდა გადაადგილდეს ქალაქიდან ქალაქში და ესტუმროს ამ ქალაქებში არსებულ ატრაქციონებს.

ტაივანში n ქალაქია, რომლებიც განლაგებულია ერთი მაგისტრალის გასწვრივ. ქალაქები გადანომრილია მიმდევრობით 0-დან $(n - 1)$ -მდე. i -ური ქალაქისათვის $(0 < i < n - 1)$ მეზობელი ქალაქებია $(i - 1)$ -ე და $(i + 1)$ -ე ქალაქები. ქალაქ 0-ის ერთადერთი მეზობელია ქალაქი 1, ქალაქ $(n - 1)$ -ის ერთადერთი მეზობელია ქალაქი $n - 2$.

ყოველ ქალაქში არის ატრაქციონების გარკვეული რაოდენობა. ჯიან-ჯი გეგმავს ესტუმროს რაც შეიძლება ბევრ ატრაქციონს. ჯიან-ჯის შერჩეული აქვს ქალაქი, საიდანაც უნდა დაიწყოს სადღესასწაულო მოგზაურობა. ყოველ დღეს ჯიან-ჯის შეუძლია შეასრულოს ორი მოქმედებიდან ერთ-ერთი:

გადაადგილდეს მეზობელ ქალაქში, ან ესტუმროს ყველა ატრაქციონს იმ ქალაქში, რომელშიც იმყოფება. ორივე მოქმედების ერთდროულად შესრულება მას არ შეუძლია. ჯიან-ჯის დევიზია აგრეთვე, *არასდროს არ ესტუმროს ერთი და იგივე ქალაქში ატრაქციონებს ორჯერ*, თუნდაც მრავალჯერ დარჩეს ამ ქალაქში. დაეხმარეთ ჯიან-ჯის ისე დაგეგმოს თავისი სადღესასწაულო მოგზაურობა, რომ ესტუმროს რაც შეიძლება ბევრ განსხვავებულ ატრაქციონს.

მაგალითი

დავუშვათ, ჯიან-ჯის აქვს 7-დღიანი დღესასწაულები, მოცემულია 5 ქალაქი (ჩამოთვლილია ქვემოთ ცხრილში) და იწყებს მოგზაურობას მე-2 ქალაქიდან. პირველ დღეს ჯიან-ჯი ესტუმრება მე-2 ქალაქში 20 ატრაქციონს. მეორე დღეს ჯიან-ჯი გაემგზავრება მე-3 ქალაქში, მე-3 დღეს ესტუმრება 30 ატრაქციონს მე-3 ქალაქში. შემდეგ 3 დღეს ჯიან-ჯი მოანდომებს მე-3-დან 0-ვან ქალაქში გადაადგილებას, შემდეგ დღეს ესტუმრება 10 ატრაქციონს 0-ვან ქალაქში (მეშვიდე დღეს). სულ ჯიან-ჯი ესტუმრება $20 + 30 + 10 = 60$ ატრაქციონს. ეს არის მაქსიმალური რაოდენობა ატრაქციონებისა, რომლებსაც ესტუმრება ჯიან-ჯი 7 დღის განმავლობაში, როცა მოგზაურობას დაიწყებს მე-2 ქალაქიდან.

ქალაქი	ატრაქციონების რაოდენობა
0	10
1	2
2	20
3	30
4	1

დღე	მოქმედება
1	ესტუმრება ატრაქციონებს მე-2 ქალაქში
2	გაემგზავრება მე-2 ქალაქიდან მე-3 -ში
3	ესტუმრება ატრაქციონებს მე-3 ქალაქში
4	გაემგზავრება მე-3 ქალაქიდან მე-2 -ში
5	გაემგზავრება მე-2 ქალაქიდან 1-ლში
6	გაემგზავრება 1-ლი ქალაქიდან 0-ვანში
7	ესტუმრება ატრაქციონებს 0-ვან ქალაქში

ამოცანის დასმა

გთხოვთ გაუკეთოთ რეალიზაცია ფუნქციას `findMaxAttraction`. იგი ითვლის ატრაქციონების მაქსიმალურ რაოდენობას, რომლებსაც შეუძლია ესტუმროს ჯიან-ჯი.

- `findMaxAttraction(n, start, d, attraction)`
 - `n`: ქალაქების რაოდენობა.
 - `start`: საწყისი ქალაქის ნომერი.
 - `d`: სადღესასწაულო დღეების რაოდენობა.
 - `attraction`: n სიგრძის მასივი; `attraction[i]` არის ატრაქციონების რაოდენობა i -ურ ქალაქში, სადაც $0 \leq i \leq n - 1$.
 - ფუნქცია დაგვიბრუნებს ატრაქციონების მაქსიმალურ რაოდენობას, რომლებსაც შეუძლია ესტუმროს ჯიან-ჯი.

ქვეამოცანები

ყველა ქვეამოცანაში $0 \leq d \leq 2n + \lfloor n/2 \rfloor$.

დამატებითი შეზღუდვები:

ქვეამოცანა	ქულა	n	ატრაქციონების რაოდენობა ქალაქში (t)	საწყისი ქალაქი
1	7	$2 \leq n \leq 20$	$0 \leq t \leq 1,000,000,000$	შეზღუდვა არ არის
2	23	$2 \leq n \leq 100,000$	$0 \leq t \leq 100$	ქალაქი 0
3	17	$2 \leq n \leq 3,000$	$0 \leq t \leq 1,000,000,000$	შეზღუდვა არ არის
4	53	$2 \leq n \leq 100,000$	$0 \leq t \leq 1,000,000,000$	შეზღუდვა არ არის

რეალიზაციის დეტალები

თქვენ უნდა გააგზავნოთ ერ-ერთი ფაილი `holiday.c`, `holiday.cpp` ან `holiday.pas`. ფაილში რეალიზება უნდა გაუკეთოთ ზემოთ აღწერილ ქვეპროგრამას შესაბამისი შაბლონით. აგრეთვე უნდა ჩართოთ ფაილი `holiday.h` (C/C++)-ში რეალიზაციისას.

მიაქციეთ ყურადღება, რომ შედეგი შეიძლება იყოს დიდი და `findMaxAttraction`-ის მიერ უნდა დაბრუნდეს 64 ბიტიანი მთელი რიცხვი.

C/C++ პროგრამა

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

Pascal პროგრამა

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

grader-ის მაგალითი

grader-ის მაგალითში შემავალი მონაცემების შეტანა ხდება შემდეგი ფორმატით:

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

grader-ის მაგალითმა უნდა დაბეჭდოს `findMaxAttraction`-ის მიერ დაბრუნებული მნიშვნელობა.