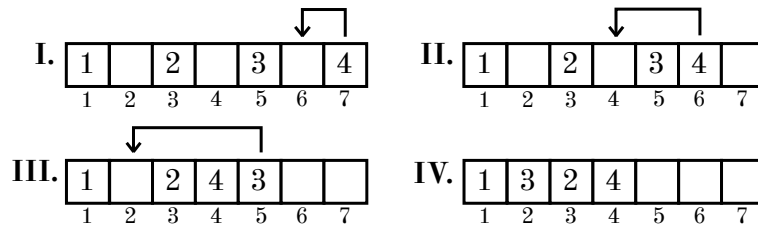


## Problem Aivazovsky. Прескачане в масив

Input file: `input.txt` or standard input  
Output file: `output.txt` or standard output  
Time limit: 2 seconds  
Memory limit: 512 megabytes

Дима е начинаещ програмист. В хода на своята работа той редовно трябва да прави едно и също действие: да премахва всеки втори елемент от масив. На Дима му омръзва тази проста задача и в един прекрасен ден той измисля оригинален алгоритъм.

Алгоритъмът на Дима е следния: Нека първоначално в масива има  $n$  числа от 1 до  $n$ , като числото  $i$  се намира в клетката с индекс  $2i - 1$  (номерацията на елементите в масива започва от единица), а останалите клетки на масива са празни. След това, на всяка стъпка, Дима избира непразната клетка на масива с максимален индекс и премества записаното в нея число в най-близката празна клетка наляво от избраната. Процесът продължава дотогава, докато всички  $n$  числа се разположат в първите  $n$  клетки на масива. Например, ако  $n = 4$ , съдържанието на масива се променя по следния начин:



Напишете програма, която определя, кое число се намира в клетката с номер  $x$  ( $1 \leq x \leq n$ ) след приключване работата на алгоритъма на Дима.

### Input

На първия ред на стандартния вход се въвеждат целите числа  $n$  и  $q$  ( $1 \leq n \leq 10^{18}$ ,  $1 \leq q \leq 200\,000$ ) — броя на елементите в масива и броя на въпросите, на които е необходимо да се даде отговор. На всеки от следващите  $q$  реда се въвежда цяло число  $x_i$  ( $1 \leq x_i \leq n$ ) — номерата на клетките, за които е необходимо да се определи съдържанието на масива след завършване работата на алгоритъма.

### Output

На всичките  $q$  въпроса изведете едно цяло число — стойността, която ще получи указаната клетка на масива след завършване работата на алгоритъма на Дима.

### Example

input	output
4 3	3
2	2
3	4
4	

### Scoring

Тестовите на тази задача се състоят от три групи. Точките за всяка група се дават само при преминаване на всички тестове от групата и всички тестове от предходните групи.

Группа	Баллы	Дополнительные ограничения		Комментарий
		$n$	$q$	
0	0	–	–	Тест из условия
1	31	$n \leq 1000$	$q \leq 1000$	–
2	29	$n \leq 200\,000$	–	–
3	40	–	–	–

## Problem Van Gogh. Астрономия

Input file:           input.txt or standard input  
Output file:          output.txt or standard output  
Time limit:           5 seconds  
Memory limit:        512 megabytes

18 година от нашата ера. Известният астроном Филон Берляндски публикува трактат "За устройството на небесния свод в който разказва за невероятно явление, забелязано от него, докато наблюдава звездите. Веднъж, през нощта Филон видял на безоблачното небе  $2n$  звезди и Луната. Удивително е, че звездите можело да бъдат мислено разбити на двойки, така че всяка права, преминаваща през центъра на двете звезди от двойката да премине също и през центъра на Луната. При това всички такива прави са различни. Филон старателно отбелязал даденото явление на картата на звездното небе, в която била въведена правоъгълна координатна система, и открил, че центровете на всичките звезди и центърът на Луната се намират в точки с целочислени координати. Тъй като Филон считал, че Земята и Луната са плоски, координатната система на картата била двумерна. Координатната система била избрана от астронома така, че координатите на всички обекти, включително Луната, по модул не надвишават  $10^6$ . Освен това, никои два обекта (две звезди или звезда и Луна) не са в една точка.

Освен картата на звездното небе Филон Берляндски записал в своя трактат предсказание, че след 2000 години звездите на небето ще се върнат в същото положение, а на мястото на Луната ще се появи огромна комета, която ще унищожи Земята.

2018 година от нашата ера. В ръцете ви попада трактата на Филон Берляндски, и вие с ужас откривате, че звездите на небето се намират в същото положение като преди 2000 години! За съжаление, времето не е пощадило картата на астронома, затова на нея са останали отбелязани само точките, които съответстват на центровете на звездите, и не са запазени никакви описания, по какъв начин са били разбити точките по двойки, така че всички прекарани през тези двойки прави, да преминат и през центъра на Луната. Още по-лошо, на картата се е заличила и точката, съответстваща на центъра на Луната. За да разберем от къде ще долети кометата и за да спасим човечеството от неминуема гибел, е необходимо спешно да възстановим някаква подходяща позиция на центъра на Луната.

### Input

На първия ред на стандартния вход се въвежда цяло число  $n$  ( $2 \leq n \leq 2600$ ) — броя двойки звезди, които астрономът е видял на небето.

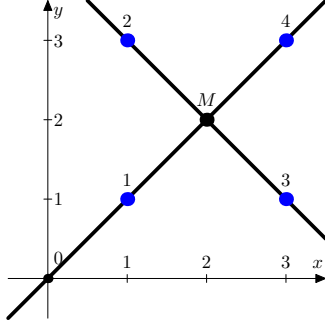
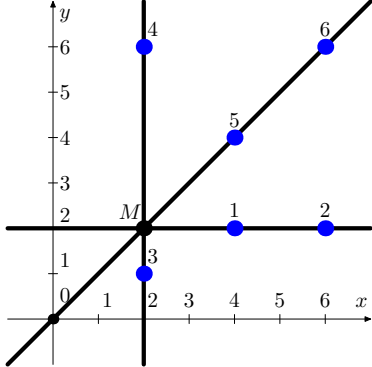
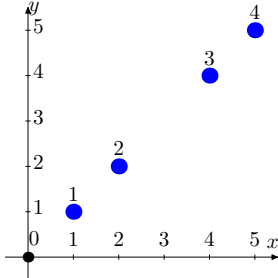
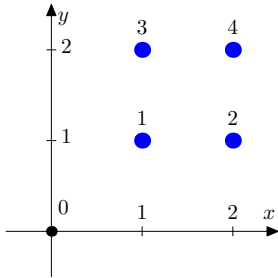
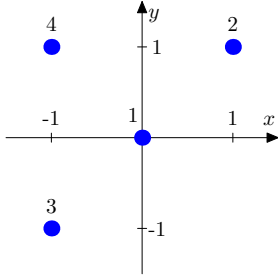
В следващите  $2n$  реда съдържат двойки цели числа  $x_i, y_i$  ( $-10^6 \leq x_i, y_i \leq 10^6$ ) — координатите на центровете на звездите на картата. Обърнете внимание, че звездите са указани в произволен ред, който в никой случай не е свързан с това, как Филон Берляндски ги е разделил по двойки. Центровете на никои две звезди не се намират в една точка.

### Output

Ако астрономът е допуснал грешка, и няма начин да се разбият всичките точки на двойки, по такъв начин, че всички построени при това прави да са различни и да се пресичат в една точка с цели координати, различна от центровете на всички звезди, изведете "No" (без кавичките) на единствения ред на стандартния изход.

В противен случай на първия ред изведете "Yes" (без кавички). На втория ред изведете двойка цели числа  $x, y$  ( $|x|, |y| \leq 10^6$ ) — координатите на точката, в която се намира центърът на Луната във вашето решение. Ако подходящите точки са няколко, то изведете коя да е от тях. Обърнете внимание, че изведената точка не може да съвпада с центъра на нито една от звездите.

## Examples

input	output	illustration
2 1 1 1 3 3 1 3 3	Yes 2 2	
3 4 2 6 2 2 1 2 6 4 4 6 6	Yes 2 2	
2 1 1 2 2 4 4 5 5	No	
2 1 1 2 1 1 2 2 2	No	
2 0 0 1 1 -1 -1 -1 1	No	

## Note

В четвъртия тест от условието, центърът на Луната може да се разположи само в точката  $(1.5, 1.5)$ , но тази точка не е с цели координати, затова няма отговор.

В петия тест от условието е невъзможно да се намери подходяща точка, която при това да не е център на звезда.

## Scoring

Тестовите към тази задача се състоят от осем групи. Точките за всяка група се получават само при преминаване на всички тестове в групата и всички тестове от предходните групи.

Подзадача	Точки	Допълнителни ограничения	Коментarii
		$n$	
0	0	—	Тестове от условието
1	9	$n \leq 2$	—
2	10	$n \leq 5$	—
3	9	$n \leq 25$	—
4	9	$n \leq 200$	—
5	10	$n \leq 500$	—
6	11	$n \leq 1000$	—
7	10	$n \leq 1500$	<b>Offline-проверка</b>
8	11	$n \leq 2000$	<b>Offline-проверка</b>
9	11	$n \leq 2300$	<b>Offline-проверка</b>
10	10	—	<b>Offline-проверка</b>

## Problem Raphael. ООШП

Input file:           input.txt or standard input  
Output file:         output.txt or standard output  
Time limit:           2 seconds  
Memory limit:       512 megabytes

Открито Обединение на Шиномонтажници-Перфекционисти (ООШП) е организация, обединяваща  $n$  шиномонтажници от град  $N$ . Всички шиномонтажници в тази организация са номерирани с числата от 1 до  $n$  в реда на тяхното постъпване в ООШП и са обединени в една дървовидна йерархия по такъв начин, че шиномонтажник номер 1 е ръководител на организацията, а всеки друг шиномонтажник  $i$  има *непосредствен началник*  $p_i$ , който задължително е с по-малък номер. Ще казваме, че шиномонтажник  $v$  е *началник* на шиномонтажник  $u$ , ако  $v$  се среща в редицата от непосредствени началници от  $u$  до 1, т.е. в последователността  $p_u, p_{p_u}$  и т.н. Шиномонтажника  $u$  в такъв случай се нарича *подчинен* на шиномонтажника  $v$ .

Тъй като всички членове на ООШП са още и перфекционисти, то в хода на работата между тях често възникват спорове. Приемаме, че споровете могат да възникнат само между двама шиномонтажника, нито един, от които не е подчинен на другия. За разрешаване на спора те отиват при своя *най-близък общ началник*, т.е. към шиномонтажника с максимален номер, който е началник на всеки от спорещите. Всеки шиномонтажник, освен първия, притежава някакво *ниво на перфекционизъм*, изразено с цяло число  $c_i$ . *Нажеженост* на спора се нарича сборът от нивата на перфекционизма на двамата, участващи в него шиномонтажници. Накрая, *конфликтност* на работния ден се нарича сборът от нажеженостите на всички спорове, възникнали през този ден. В края на работния ден шиномонтажник  $v$  смята себе си за *ефективен ръководител*, ако през този ден е помогнал за разрешаването на поне един спор на всеки от своите подчинени. Формално казано, това означава, че за всеки шиномонтажник  $u$ , който е подчинен на  $v$ , съществува такъв шиномонтажник  $w$ , че  $u$  и  $w$  са имали конфликт през деня, а  $v$  е бил най-близкият общ началник на  $u$  и  $w$ . В частност, всеки шиномонтажник, който няма подчинени, автоматически решава, че ефективен ръководител.

Вие работите като програмист в ООШП и се познавате с всички шиномонтажници в организацията. Излизайки днес от работа, всеки шиномонтажник в компанията под секрет ви е съобщил, че като цяло за днешния ден смята себе си за ефективен ръководител. Вие знаете иерархията на шиномонтажниците в ООШП, но не знаете, какви спорове са възникнали между тях през деня. Сега на вас ви е интересно, каква минимална стойност може да има конфликтността на днешния ден, при условие, че всеки шиномонтажник наистина е бил ефективен ръководител през деня.

### Input

На първия ред на стандартния вход се въвежда цяло число  $n$  ( $3 \leq n \leq 200\,000$ ) — броя шиномонтажници в ООШП. На втория ред на стандартния вход се въвеждат  $n - 1$  цели числа  $p_2, p_3, \dots, p_n$  ( $1 \leq p_i < i$ ), където  $p_i$  съответства на номера на шиномонтажника, който е началник на шиномонтажника с номер  $i$ . На третия ред се въвеждат  $n - 1$  цели числа  $c_2, c_3, \dots, c_n$  ( $1 \leq c_i \leq 10^6$ ), където  $c_i$  е нивото на перфекционизма на  $i$ -тия шиномонтажник.

Гарантирано е, че при зададената структура на йерархията може да има такава ситуация, че в края на работния ден всеки шиномонтажник да смята себе си за ефективен ръководител.

### Output

Изведете минималната възможна стойност на конфликтност за днешния ден.

## Examples

input	output
5 1 2 2 1 1 1 1 1	8
6 1 1 1 4 4 1 2 3 4 5	25

## Note

Да разгледаме първия тест от условието. За да се достигне указаната стойност на конфликтност на деня, трябва през деня да са възникнали спорове в следните двойки шиномонтажници (2, 5), (3, 4), (3, 5) и (4, 5).

- Шиномонтажници 3, 4 и 5 автоматично смятат себе си за ефективни ръководители, тъй като нямат подчинени.
- Шиномонтажник 2 смята себе си за ефективен ръководител, защото е помогнал на шиномонтажник 3 в спора с шиномонтажник 4, а на шиномонтажник 4 в спора с шиномонтажник 3.
- Шиномонтажник 1 смята себе си за ефективен ръководител, тъй като е помогнал на шиномонтажници 2, 3 и 4 да изгладят конфликтите си с шиномонтажник 5, а на шиномонтажник 5 той е помогнал да разреши цели три конфликта.

Нажежеността на всеки спор е равна на  $2 = 1 + 1$ , затова стойността на конфликтността през този ден е 8.

Във втория пример (който се включва в ограниченията на втората и четвъртата, но не и на първата и третата групи тестове) оптимално решение може да се получи при спорове в двойките (2, 4), (2, 5), (3, 6) и (5, 6). Стойността на конфликтността за деня при този сценарий е  $(1 + 3) + (1 + 4) + (2 + 5) + (4 + 5) = 25$ . Указаният набор двойки не е единственият начин да се получи минимална стойност на конфликтност.

## Scoring

Тестовите към тази задача се състоят от четири групи. Точките за всяка група се получават само при преминаване на всички тестове в групата и всички тестове от някои от предходните групи.

**Offline-проверка** означава, че резултатите от тестването на вашето решение за дадената група ще станат известни едва след завършване на състезанието.

Група	Точки	Допълнителни ограничения		Необх. групи	Коментари
		$n$	$c_i$		
0	0	–	–	–	Тестовите от условието
1	25	$n \leq 2000$	$c_i = 1$	–	–
2	26	$n \leq 2000$	–	0–1	–
3	25	–	$c_i = 1$	1	<b>Offline-проверка</b>
4	24	–	–	0–3	<b>Offline-проверка</b>

## Problem Leonardo. Културен контакт

Input file: `input.txt` or standard input  
Output file: `output.txt` or standard output  
Time limit: 1 second  
Memory limit: **128 megabytes**

В началото на XVIII век група европейски изследователи пристигнала на остров, населен от група племена, които никога не са имали контакт с представители на европейска цивилизация.

За успешното създаване на контакт с аборигените ръководителят на групата планира да направи подарък на вожд на всяко срещнато племе. За тази цел той носи дълъг гердан от скъпоценни камъни. Представяме гердана като низ  $s$ , състоящ се от малки букви на английската азбука, където всяка буква означава типа на скъпоценният камък на съответната позиция. Изследователите имат намерение да разрежат гердана на няколко части, след което връчват точно една част на вожд на всяко следващо племе. Ръководителя на изследователите решил да раздели гердана на части, съгласно следните правила:

- За да не се губи много време за разрязването, всяка част трябва да се състои от група съседни камъни от гердана, т.е. поднизове на низа  $s$ .
- Всички скъпоценни камъни трябва да бъдат използвани, т.е. всеки камък трябва да бъде включен в точно една част от гердана.
- Тъй като изследователите не знаят какви камъни предпочитат аборигените, те искат, всеки вожд да получи един и същ комплект камъни, без да се отчита реда. С други думи, за всеки тип камък, броят на камъните от този тип, трябва да бъде еднакъв във всяка от частите на гердана.
- Изследователите не знаят колко племена живеят на острова, затова броят на подготвените части от гердана, трябва да бъде максимален.

Помогнете на ръководителя да определи максималният брой части, които могат да се получат.

### Input

На първият ред на стандартният вход е записан непразен низ  $s$ , състоящ се от малките букви на английската азбука. Дължината на низа  $s$  не надминава  $5 \cdot 10^6$  символа.

### Output

Да се изведе едно число – максималният брой части, на които изследователите могат да разрежат гердана, който имат, без да нарушават нито едно от условията, формулирани от ръководителя на групата.

### Examples

input	output
abbabbbab	3
aabb	1

### Note

В първият пример изследователите могат да разделят гердана `abbabbbab` на части `abb`, `abb`, `bab`, тогава при всяка среща с племето вожд ще получи по един камък от тип `a` и по два камъка от тип `b`.

Във втория пример, низът не е възможно да се раздели на повече от една част, поради зададените условия.



## Scoring

Тестовете в тази задача се състоят от пет групи. Точките за всяка група се дават само при вярно решение на всички тестове от групата и всички тестове от предходните групи. **Offline-проверка** означава, че резултати от тестването на вашето решение на дадена група ще станат достъпни, само след края на състезанието.

Група	Баллы	Дополнительные ограничения	Необх. группы	Комментарий
0	0	–	–	Тесты из условия
1	15	$ s  \leq 100$	0	–
2	15	$ s  \leq 1000$	0, 1	–
3	21	$ s  \leq 500\,000$	0–2	–
4	20	Строка состоит только из символов <b>a</b> и <b>b</b>	0	–
5	29	–	0–4	<b>Offline-проверка</b>