

Problem Barcelona. Медианен планински хребет

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

Берляндия — огромна страна с разнообразна география. Една от най-големите природни забележителности на Берляндия е «Медианният планински хребет». Този планински хребет представлява n на брой, поредни планински върха, разположени на една права, номерирани последователно от 1 до n . Височината на i -тия планински връх е равна на a_i .

«Медианният планински хребет» е известен с това, че при него ежедневно се случва така нареченото *изравняване на планинските върхове*. В моментът на изравняването, едновременно, за всеки връх с номер от 2 до $n - 1$, височината на даден връх става равна на медианата на съответният връх и двата му съседни върха. Иначе казано, ако преди изравняването, височините са били равни на b_i , то новите височини a_i са следните: $a_1 = b_1$, $a_n = b_n$, а за всяко i от 2 до $n - 1$ $a_i = \text{median}(b_{i-1}, b_i, b_{i+1})$. Медиана на три числа се нарича второто подред число, ако сортираме в нарастващ ред трите числа. Например, $\text{median}(5, 1, 2) = 2$, а $\text{median}(4, 2, 4) = 4$.

Наскоро Берляндските учени доказаха, че за произволни стойности на височините на върховете, процесът на изравняване рано или късно се стабилизира, т.е. в определен момент височината на върховете престава да се променя след изравняване. Правителството на Берляндия иска да разбере след колко години ще се случи това, т.е., намерете величината c — броят на изравняванията, които ще се случат, при които поне един връх ще промени своята височина. Помогнете на учените да решат тази важна задача!

Обърнете внимание, че някои от тестовите групи, освен стойността на величината c , вие трябва да определите и височината на върховете след c на брой изравнявания, т.е. трябва да се определи височината, на която върховете ще останат завинаги.

Input

На първият ред са дадени целите числа n и t ($1 \leq n \leq 500\,000$, $0 \leq t \leq 1$) — броя на върховете и параметър, който определя, трябва ли да се определят крайните височини на върховете.

Вторият ред съдържа целите числа $a_1, a_2, a_3, \dots, a_n$ ($1 \leq a_i \leq 10^9$) — текущите височини на върховете.

Output

На първият ред изведете едно цяло число c — броят на изравняванията на върховете, при които височината на поне един връх се променя.

Ако $t = 1$, на вторият ред изведете n на брой цели числа — крайните височини на върховете след c на брой изравнявания.

Examples

input	output
5 1 1 2 1 2 1	2 1 1 1 1 1
6 1 1 3 2 5 4 6	1 1 2 3 4 5 6
6 0 1 1 2 2 1 1	0

Note

В първият пример височината на позиции 1 и 5 не се променя. Тъй като медианата на числата 1, 2, 1 е равна на 1, то на позиции 2 и 4, след първото изравняване, се оказва числото 1, и тъй като медианата на числата 2, 1, 2 е равна на 2, то на позиция 3, след първото изравняване, се оказва числото 2. Накрая, след първото изравняване, планинските върхове имат височини 1, 1, 2, 1, 1. След второто изравняване височината се установява на 1, 1, 1, 1, 1 и по-нататък няма да се променя, съответно, общо са били 2 на брой променения на височината.

В третият пример, след изравняване, при нито един връх няма да настъпи промяна във височината, следователно броят на промените, при които височината се променя е равен на 0. Тъй като $t = 0$, няма нужда да се изведат крайните стойности на височините.

Scoring

Тестовите към тази задача са групирани в шест групи. Точки за всяка група ще се дават само при преминаване на всички тестове от съответната група, както и при преминаване на всички тестове на **необходимите** групи. **Offline-проверка** означава, че резултатите от тестването на вашето решение за дадената група ще станат известни след приключване на състезанието.

Група	Точки	Допълнителни ограничения			Необх. групи	Коментари
		n	a_i	t		
0	0	–	–	–	–	Тестовите от условието.
1	19	$n \leq 1000$	–	–	0	Гарантирано е, че $c \leq 10\,000$.
2	24	–	$a_i \leq 2$	–	–	
3	14	–	$a_i \leq 100$	$t = 0$	–	
4	14	–	$a_i \leq 100$	–	0, 2, 3	
5	14	–	–	$t = 0$	3	Offline-проверка.
6	15	–	–	–	0, 1, 2, 3, 4, 5	Offline-проверка.

Problem Wimbledon. Интересни конкурси

Input file: input.txt or standard input
Output file: output.txt or standard output
Time limit: 1 second
Memory limit: 512 megabytes

Учителката на Димитър дала следната задача на класа му. Тя накарала всеки ученик да измисли последователност от отварящи и затварящи скоби с произволна дължина. След това учениците подред показали измислените от тях последователности от скоби. Когато наближил реда на Дима, той изведнъж разбрал, че при всички негови съчученици се получили правилни последователности от скоби, а дали неговата е такава, той не разбрал.

Дима подозира, че просто не е чул думата «правилна» в условието на задачата и искал много бързо да поправи ситуацията. Той решил да изпълни произволен брой пъти операцията *разместване*. За изпълнението на едно разместване Дима избира подниз от последователността и по произволен начин пренарежда символите в него. Такава операция се изпълнява за l наносекунди, където l е дължината на подниза, който Дима си е избрал. Лесно е да се забележи, че в резултат на разместванията броят на отварящите и затварящите скоби не се променя.

Съвсем скоро ще дойде реда на Дима да каже своята последователност, затова той се обръща за помощ към вас. Помогнете му да определи най-малкото време, за което може да направи последователността си правилна, или определете, че това е невъзможно.

Input

На първия ред на входа е зададено цялото число n ($1 \leq n \leq 10^6$) — дължината на последователността на Дима.

Втория ред съдържа низ с дължина n , състоящ се само от символите «(» и «)».

Output

Изведете едно число — най-малкия брой наносекунди, за които последователността ще стане правилна. Изведете «-1», ако е невъзможно последователността да стане правилна.

Examples

input	output
8)) ((() ((6
3 ((-1

Note

Напомняме, че една последователност от скоби е правилна, ако при поставяне на символите «+» и «1» измежду тях ще получим коректен математически израз. Например, последователностите «(())()», «()» и «((()())» са правилни, а «()», «(())» и «((())()» — не са.

В първия пример отначало разместваме подниз от първи до четвърти символ и го заместваме с «()()» — получаваме последователността «()()()()». След това разместваме скобите в подниза от седми до осми символ, заместваме го с «()». Получаваме правилна последователност от скоби «()()()()». Тези действия се извършват за $4 + 2 = 6$ наносекунди.

Scoring

Тестовете към задачата се състоят от две групи. Точките за всяка група се дават само, ако са верни всички тестови примери в групата и всички тестови примери в **предходните** групи.

Група	Точки	Допълнителни ограничения	Коментар
		n	
0	0	–	Тестове от условието
1	50	$n \leq 1000$	
2	50	–	

Problem Zermatt. Разработка на тарифи

Input file: input.txt or standard input
Output file: output.txt or standard output
Time limit: 6 seconds
Memory limit: 512 megabytes

Кметът на град М. решил през 2020 година да пусне няколко нови линии на метрото. Тъй като бюджетът на града е силно ограничен, той взел решение да не копае нови тунели, а да използва вече съществуващата подземна мрежа.

Тунелната система на града М. се състои от n станции на метрото. Станциите са свързани помежду си с двупосочни тунели, които са точно $n - 1$. Между кои да е две станции v и u има точно един прост път. Всяка линия на метрото, която иска да създаде кмета, е прост път от станция a_i до станция b_i . Линиите могат да се пресичат, т.е. да имат общи станции или тунели. Но, още не е решено, в коя от двете страни ще вървят влаковете по всяка от линиите. А именно, на пътя между станциите a_i и b_i влаковете ще вървят или от станция a_i към станция b_i , или от станция b_i към станция a_i , но само в едната посока.

В града М действа особена система от тарифи. На всяка станция се присвоява цяло положително число c_i — наречено тарифна зона на станцията, а стойността на придвижването от станция v до станция u се пресмята като $c_u - c_v$ бурли. Разбира се, такова придвижване е разрешено, само ако има линия на метрото, по което влаковете вървят от v към u . Кметът на града не иска, между някои две станции от една линия стойността на придвижването да бъде отрицателна, затова било взето решение да се изберат направления на движенията на влаковете по линиите и да се изменят тарифните зони на всички станции в града по такъв начин, че за всяка линия тарифните зони на станциите строго нарастват, ако ги разглеждаме в посока на движение на влаковете.

Кметът иска отначало да присвои на всяка станция тарифна зона, а след това да избере направления на всички линии в метрото, така че за всяка линия тарифните зони на станциите строго нарастват. Във връзка с предстоящия празник на града, от всички възможни варианти за определянето на тарифните зони кметът иска да избере такъв, за който максималното c_i да е колкото може по-малко. Помогнете на кмета да състави ново разпределение, или кажете, че това е невъзможно. Обърнете внимание, че от вас се изисква само да определите тарифните зони по оптимален начин, а направленията на всички линии не се изискват. Така, вашето решение се приема за вярно, ако съществува начин да се избере направление на движението на всяка линия така, че за всички линии тарифните зони строго нарастват в посока на движението на влака.

Обърнете внимание, че в някои групи тестове не е нужно да се минимизира отговора, а се изисква само да се определи дали е възможно да се зададат тарифните зони по този начин.

Input

Първия ред съдържа цяло число n, m, t ($2 \leq n, \leq 500\,000, 1 \leq m \leq 500\,000, 0 \leq t \leq 1$) — броя станции в града, броя линии в метрото, и един параметър на теста t . Ако $t = 0$, минимизирането на отговора не е задължително. Ако $t = 1$, от вас се изисква да минимизирате стойността на тарифната зона на най-скъпата станция.

Всеки от следващите $n - 1$ реда описват поредния тунел на метрото. Всеки тунел се задава с целите числа v_i, u_i ($1 \leq v_i, u_i \leq n, v_i \neq u_i$). Гарантирано е, че между кои да е две станции има точно по един прост път.

Всеки от следващите m реда описват поредната линия на метрото. Всяка линия се задава с целите числа a_i, b_i ($1 \leq a_i, b_i \leq n, a_i \neq b_i$).

Output

На първия ред изведете цяло число k — максималната стойност на тарифната зона. Ако параметърът на теста t е равен на 0, от вас не се изисква да се минимизира k . Ако $t = 1$, то k трябва да бъде

минималното възможно.

На следващия ред изведете числата c_1, c_2, \dots, c_n ($1 \leq c_i \leq k$) — тарифните зони на станциите.

Ако съществуват няколко отговора, изведете кой да е от тях. Ако не съществува нито един начин да са определят тарифните зони, изведете «-1».

Examples

input	output
3 1 1 1 2 2 3 1 3	3 1 2 3
4 3 0 1 2 1 3 1 4 2 3 2 4 3 4	-1

Note

В първия пример линията $1 \rightarrow 3$ преминава през станциите в последователност 1, 2, 3. При такава последователност на посещение на станциите техните тарифни зони нарастват. Тъй като на това разклонение са разположени 3 станции, то ще ни трябва минимум 3 тарифни зони. В този случай отговорът 1, 2, 3 е оптимален.

Във втория пример нито едно разпределение на тарифните зони не се съгласува с линиите на метрото.

Scoring

Тестовите към тази задача са разпределени в десет групи. Точките за всяка група се дават само при преминаване на всички тестове от групата и всички тестове от **необходимите** групи. **Offline-проверка** означава, че резултатите от тестването на вашето решение на дадената група ще станат достъпни само след завършване на състезанието.

В групите 4 и 5 не съществува тунел, принадлежащ на две линии на метрото.

Дължина на линията ще наречем броя тунели, които и принадлежат.

XIV Moscow Open Olympiad in Informatics, day 1
Russia, Moscow, March 6

Група	Точки	Допълнителни ограничения		Необх. групи	Коментари
		n, m	t		
0	0	–	–	–	Тестовете от условието
1	6	$n, m \leq 8$	–	0	
2	10	$n, m \leq 15$	–	0, 1	
3	15	$n, m \leq 100$	–	0, 1, 2	
4	11	$n, m \leq 100\,000$	–	–	Няма общи тунели.
5	10	–	–	4	Няма общи тунели.
6	8	–	–	–	Всички тунели имат общ край.
7	10	$n, m \leq 100\,000$	$t = 0$	–	Общата дължина на тунелите е до 100 000.
8	6	–	$t = 0$	7	
9	14	$n, m \leq 100\,000$	–	0, 1, 2, 3, 4, 7	
10	10	–	–	0 – 9	Offline-проверка.

Problem Dakar. Двоен палиндром

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

Ваня работи във фабрика за палиндроми. Във фабриката има детайл — низ s с дължина n , състоящ се от малки английски букви, от които Ваня може да изреже всеки подниз за продажба. Да припомним, че *палиндром* — това е низ, който се чете по един и същ начин както от ляво на дясно, така и от дясно на ляво.

Обикновено палиндромите са омръзнали на всички и никой не ги купува, затова той решил да произведе двойни палиндроми. *Двоен палиндром* — това е низ, който се явява слепване на два палиндрома с **равни** дължини. Например, низовете «aabb», «aaaa» са двойни палиндроми, а низовете «abba» и «aaaabb» не са.

Ваня се чуди колко са начините за изрязване на двоен палиндром от s , с други думи, колко са двойките (l, r) , такива че поднизът $s_l s_{l+1} \dots s_r$ е двоен палиндром. Моля, помогнете на Ваня да намери отговор на този въпрос.

Input

На първия ред е зададено цялото число n ($1 \leq n \leq 500\,000$) — дължината на низа s . На втория ред е зададен низ s , състоящ се от малки английски букви.

Output

Изведете едно цяло число — търсения брой поднизове, които се явяват двойни палиндроми.

Examples

input	output
6 abacac	6
5 aaaaa	6

Note

В първия пример двойни палиндроми са 5 подниза с дължина 2 («ab», «ba», «ac», «ca» и «ac»), а също и целия низ («abacac»).

Scoring

Тестовите на тази задача са групирани в три подзадачи. Точките за всяка подзадача се получават само при преминаване на всички тестове за тази подзадача и всички тестове на **предишните** подзадачи.

Подзадача	Точки	Допълнителни ограничения	Коментар
		n	
0	0	–	Тестове от условието.
1	19	$n \leq 500$	
2	33	$n \leq 5000$	
3	48	–	