Awantura o czapki

Zadanie: **A**
Limit pamięci: **256 MB**
Limit czasu: **0.5 s**

<Mistrzostwa w Algorytmice i Programowaniu>

FUNDACJA ROZWOJU INFORMATYKI

Ministerstwo Cyfryzacji

Uniwersytet Wrocławski

Nadchodzi lato! W przeciwieństwie do zimy, w kulturze krasnoludków lato to czas radości, spokoju i przede wszystkim... hucznych przyjęć! Gdy tylko zaczynają się pierwsze gorące dni, w całym królestwie krasnoludków rozpoczynają się przygotowania do Krasnalfestu, corocznego festiwalu tańca, śpiewu i oczywiście programowania.

Należy jeszcze wspomnieć, że krasnoludki to naród bardzo ambitny i pracowity. Z racji tego, że Krasnalfest jest najważniejszym wydarzeniem w całym roku, nawet najmniejszy szczegół festiwalu musi zostać dopracowany do perfekcji. Dlatego też krasnoludki Bernaś i Warchlaś mają nie lada zadanie – muszą zająć się dekoracjami.

Wszystko szło im wyśmienicie do momentu, aż dotarli do kwestii tradycyjnej wystawy czapek. Bernaś od razu zaproponował ustawienie czapek od najniższej do najwyższej. Warchlaś natomiast był odmiennego zdania – według niego dużo ciekawsze byłoby ułożenie od najwyższej do najniższej. Po kilku dniach debatowania postanowili zaniechać dalszych sporów i zdecydowali się na połączenie tych dwóch pomysłów.

Ustalona przez nich kolejność czapek miała spełniać następujący warunek: ciąg $a_1, a_2, \ldots, a_N$, oznaczający wysokości kolejnych czapek, musi zawierać podciąg rosnący długości K i podciąg malejący długości L. Oczywiście ciąg $a_1, a_2, \ldots, a_N$ nazywamy rosnącym, gdy $a_1 < a_2 < \ldots < a_N$. Analogicznie ciąg jest malejący, gdy $a_1 > a_2 > \ldots > a_N$. Podciągiem nazywamy ciąg powstały w wyniku wykreślenia pewnych elementów (być może żadnego) oryginalnego ciągu.

Pozostało jedynie zlecić krasnoludzkim krawcom uszycie odpowiednich czapeczek. Pomóż Bernasiowi i Warchlasiowi złożyć zamówienie i napisz program, który dla danych liczb K i L wypisze długość najkrótszego możliwego ciągu wysokości czapek $a_1, a_2, \ldots, a_N$, zawierającego podciąg rosnący długości K i podciąg malejący długości L.

Wejście

W pierwszym i jedynym wierszu standardowego wejścia znajdują się dwie liczby całkowite dodatnie K i L ($1 \leqslant K, L \leqslant 1000$), oznaczające odpowiednio długość podciągu rosnącego i podciągu malejącego, który musi zawierać ciąg określony w zadaniu.

Wyjście

Na jedyny wiersz standardowego wyjścia powinieneś wypisać jedną liczbę całkowitą – długość najkrótszego ciągu, który zawiera podciąg rosnący długości K i podciąg malejący długości L.

Przykłady

Wejście
```
3 5
```
Wyjście
```
7
```
Przykładowym ciągiem, który spełnia wymagania w tym teście jest ciąg $1, 2, 5, 4, 3, 2, 1$. Można pokazać, że nie istnieje krótszy ciąg spełniający te wymagania.

Wejście
```
1 1
```
Wyjście
```
1
```

Wejście
```
10 10
```
Wyjście
```
19
```

Wejście	**Wyjście**
`1000 1000`	`1999`

Barduś i szachy

Zadanie: **B**
Limit pamięci: **256 MB**
Limit czasu: **1 s**

<Mistrzostwa w Algorytmice i Programowaniu>
FUNDACJA ROZWOJU INFORMATYKI
Ministerstwo Cyfryzacji
Uniwersytet Wrocławski

Krasnalfest słynie nie tylko z pysznego jedzenia i wesołych tańców, ale też i z prestiżowych zawodów szachowych. Krasnoludek Barduś, znany szachista i wielokrotny mistrz, chciałby kolejny raz obronić swój tytuł. Zdaje sobie jednak sprawę, że nie uda mu się tego osiągnąć bez regularnych treningów.

Zauważył, że podczas rozgrywki najgorzej idzie mu wykorzystywanie potencjału swoich gońców. Wymyślił więc ćwiczenie, które powinno pomóc mu poprawić ten aspekt gry. Układa on gońce jednego koloru na dwóch kwadratowych szachownicach o rozmiarze $N \times N$ i próbuje określić, bez poruszania żadnymi figurami, czy wykonując pewną (być może pustą) sekwencję poprawnych ruchów można z ustawienia na pierwszej szachownicy otrzymać ułożenie na drugiej. Goniec, tak jak w szachach, może poruszać się wyłącznie po przekątnych pól, w dowolnym kierunku, o dowolną liczbę niezajętych pól. Gońce nie mogą przeskakiwać nad innymi gońcami ani stawać na tym samym polu co inne.

Bardusiowi bardzo przydałoby się urządzenie sprawdzające poprawność jego odpowiedzi. Napisz program, który wczyta ułożenia gońców na dwóch szachownicach i wypisze, czy istnieje sekwencja ruchów, za pomocą której można z ustawienia na pierwszej szachownicy otrzymać ustawienie z drugiej szachownicy.

Wejście

W pierwszym wierszu standardowego wejścia znajduje się jedna liczba naturalna N ($1 \leqslant N \leqslant 1000$) oznaczająca długość boku szachownicy. W kolejnych N wierszach znajduje się opis pierwszej z szachownic. W każdym z wierszy znajduje się N znaków, każdy z nich to `X` oznaczający pole zajęte przez gońca lub `.` oznaczający wolne pole. W kolejnych N wierszach znajduje się analogiczny opis drugiej z szachownic.

Wyjście

W jedynym wierszu standardowego wyjścia powinien znaleźć się napis `TAK` jeśli możliwe jest wykonanie pewnej sekwencji ruchów tak, aby z jednego ułożenia gońców na szachownicy otrzymać drugie lub napis `NIE` w przeciwnym przypadku.

Przykłady

Wejście

```
4
.X.X
X...
....
..XX
....
X.X.
.X.X
...X
```

Wyjście

```
TAK
```

Wejście

```
4
....
....
.X..
....
....
....
X...
....
```

Wyjście

```
NIE
```

Czyścioch i porządki

Zadanie: **C**
Limit pamięci: **256 MB**
Limit czasu: **1 s**

Krasnoludek Czyścioch nie cierpi bałaganu. Wszystko w jego domu musi mieć swoje miejsce, na półkach panuje ład i harmonia.

Ostatnio jego uwagę przykuł jeden regał, na którym trzyma książki o okładkach w kolorze czerwonym i niebieskim. Od ostatniego generalnego sprzątania domu jego kolekcja znacznie się zwiększyła, przez co w oczach Czyściocha kolejność książek na półce wygląda bardzo nieelegancko. Postanowił więc zająć się wreszcie tym tematem.

Czyścioch chciałby ustawić książki w taki sposób, aby żadne dwie z niebieską okładką nie stały bezpośrednio obok siebie. Aby tego dokonać, może on wyciągnąć wybraną z półki książkę i wstawić ją w dowolne inne miejsce. Chciałby wykonać minimalną liczbę takich przestawień książek – Czyścioch wolałby pozostawić siły na sprzątanie reszty swojego domu.

Napisz program, który znając kolory okładek książek na półce, wypisze ile co najmniej przestawień należy wykonać, żeby uzyskać interesujące Czyściocha ustawienie. Jeżeli nie istnieje ciąg przestawień po którym powyższy warunek byłby spełniony, należy wypisać -1.

Wejście

W pierwszym wierszu wejścia znajduje się liczba N $(1 \leqslant N \leqslant 10^6)$, oznaczająca liczbę książek na półce. W drugim wierszu wejścia znajduje się ciąg liczb całkowitych $P_1, P_2, \ldots, P_N$ $(0 \leqslant P_i \leqslant 1)$, którego wyrazy oznaczają kolory kolejnych książek – 0 oznacza kolor czerwony, a 1 kolor niebieski.

Wyjście

W pierwszym wierszu wyjścia należy wypisać najmniejsza liczbę przestawień książek lub -1 jeżeli odpowiedź nie istnieje.

Przykłady

Wejście	Wyjście
5 1 0 1 0 1	0
8 1 0 1 1 1 0 0 0	2
3 1 1 1	-1

Dziwna kostka

Zadanie: **D**
Limit pamięci: **512 MB**
Limit czasu: **15 s**

Nadszedł najważniejszy dzień w roku dla krasnala Logika – jego urodziny. Jego ulubione zabawki to gry logiczne, dostaje je na każdą okazję. Tym razem otrzymał specjalną wersję kostki Rubika. Od oryginału różni się tym, że ma kształt piramidy (czworościanu foremnego). Każda ze ścian jest więc trójkątem równobocznym, dodatkowo podzielonym na 9 mniejszych trójkątów równobocznych. Logik zauważył, że dowolną ścianę można podzielić na trzy poziome warstwy, gdzie pierwsza złożona jest z jednego trójkąta, druga z trzech, a ostatnia z pięciu. Dla jednej ściany są trzy takie podziały, różniące się obrotem ściany (czyli tym, jaki wierzchołek jest tym 'górnym'). Logik rozpoczął dalsze obserwacje. Wybrał dowolne trzy ściany mające wspólny wierzchołek. Trójkąty będące w jednej warstwie na każdej z tych ścian tworzą poziom, który Logik może obrócić zgodnie albo przeciwnie do ruchu wskazówek zegara.

Logik położył piramidę na stole i ponumerował kolejno górny wierzchołek, lewy przedni wierzchołek podstawy, prawy przedni wierzchołek podstawy i tylny wierzchołek podstawy numerami 0, 1, 2 i 3. Przykładowa numeracja, zakładając że ściana czerwona jest ścianą przednią, widoczna jest na rysunku poniżej, po lewej stronie.

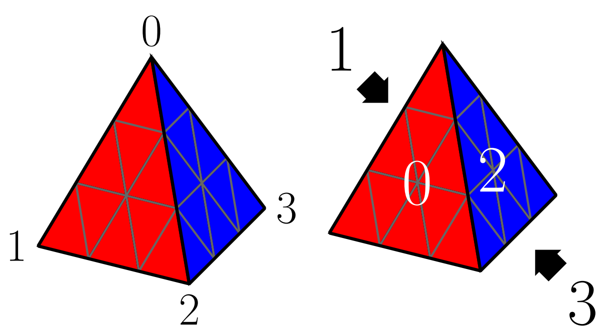
Następnie Logik ponumerował ściany, kolejno przednią, lewą, prawą i podstawę numerami 0, 1, 2 i 3. Numeracja ścian widoczna jest na poniższym rysunku, po prawej stronie.

Na koniec Logikowi zostało ponumerowanie poziomów. Od razu zdecydował, że poziom składający się z $2 \cdot k - 1$ trójkątów będzie miał numer k, dla $k \in \{1, 2, 3\}$, czyli mówiąc wprost poziomy najbliżej wierzchołka, środkowy oraz ten najbardziej odległy (którego obrót powoduje także obrót czwartej ściany wokół środka symetrii) będą miały kolejno numery 1, 2 i 3.

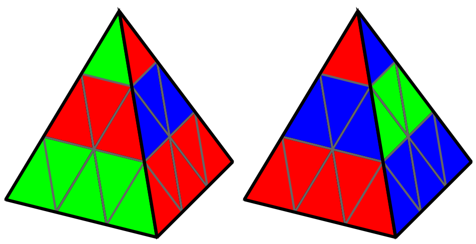
Te wszystkie informacje pozwoliły Logikowi na określenie jak będzie opisywał ruchy. Chciał je zapisać, żeby potem móc wykonać je w odwrotnej kolejności i mieć poprawnie ułożoną piramidę. Ruch polega na wybraniu wierzchołka, następnie poziomu który będzie obracał, a na końcu na wyborze kierunku obrotu: 1 – zgodnie z ruchem wskazówek zegara albo 2 – przeciwnie.

Uff, udało się, formalności ma już za sobą. Teraz może spokojnie zabrać się za układanie.

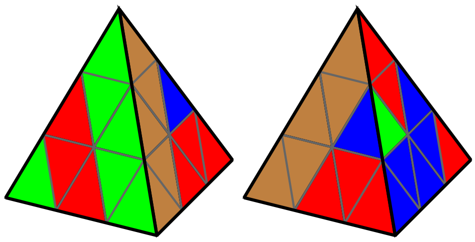
Przykładowa, poprawnie ułożona kostka, wygląda następująco (niewidoczna ściana po lewej stronie ma kolor zielony, a podstawa brązowy):



Po wykonaniu ruchu `0 2 1` piramida będzie wyglądać za to tak:



Jeżeli Logik wykona teraz ruch `2 3 1` (pamiętaj, że powyższa notacja dotyczy prawego rysunku), to piramida będzie miała następujące ułożenie:



Niestety, z uwagi na swoje wrodzone gapiostwo, Logik zgubił kartkę z opisem ruchów. Pomimo licznych prób, nie udało mu się ułożyć piramidy. Teraz prosi Ciebie o pomoc i napisanie programu, który wczyta wygląd kostki i wypisze dowolną sekwencję ruchów, po których wykonaniu piramida będzie ułożona. Piramida jest ułożona, gdy na każdej ścianie, kolor wszystkich jej dziewięciu trójkątów jest taki sam.

Wejście

Wejście składa się z dokładnie 12 wierszy, opisujących kolejno ściany $0, 1, 2, 3$. Wygląd ścian $0, 1, 2$ opisany jest, gdy piramida leży na ścianie nr 3 (czyli wierzchołek nr 0 jest na górze), natomiast wygląd ściany 3 opisany jest, gdy piramida leży na ścianie nr 0 (czyli wierzchołek nr 3 jest na górze). Opis każdej ściany składa się z 3 wierszy, opisujących kolejne warstwy tej ściany. Pierwszy wiersz zawiera jedną literę, drugi wiersz zawiera trzy litery, a trzeci zawiera pięć liter. Litery oznaczają kolory danych trójkątów, gdzie `C`, `Z`, `N` i `B` odpowiadają kolejno kolorom czerwonemu, zielonemu, niebieskiemu i brązowemu.

Wyjście

W pierwszym wierszu wyjścia powinna znaleźć się liczba kroków N ($0 \leqslant N \leqslant 1\,000\,000$). W kolejnych N wierszach powinny znaleźć się opisy kroków, które należy wykonać, aby ułożyć piramidę. Opis każdego z kroków powinien zawierać trzy liczby całkowite oddzielone pojedynczymi odstępami, będące kolejno numerem wierzchołka, numerem poziomu oraz kierunkiem obrotu (tak jak to widać we wcześniejszych przykładach).

Przykłady

Wejście

```
C
CCC
CCCCB
Z
ZZZ
ZZZZZ
N
NNN
CNNNN
B
BBB
NBBBB
```

Wyjście

```
1
2 1 2
```

Wejście

```
C
CCC
ZZZZZ
Z
ZZZ
NNNNN
N
NNN
CCCCC
B
BBB
BBBBB
```

Wyjście

```
2
0 2 2
0 1 2
```

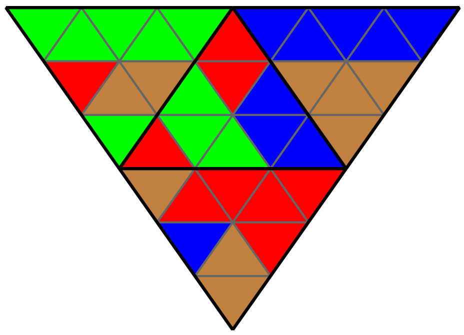
Wejście

```
C
ZCN
CZZNN
Z
ZZB
ZZCBZ
N
BNN
BBBNN
B
CBN
CCCCB
```

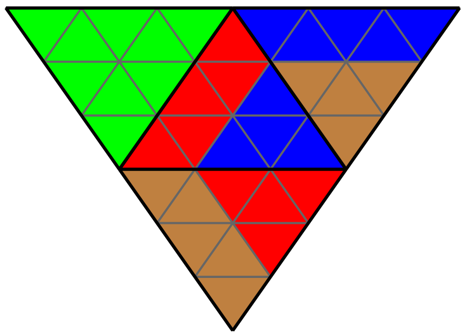
Wyjście

```
3
1 2 2
2 2 1
2 1 1
```

W trzecim teście przykładowym siatka piramidy wygląda następująco (środkowy trójkąt to ściana 0):



Siatka po wykonaniu pierwszego ruchu:



Ekonomek

Zadanie: **E**
Limit pamięci: **256 MB**
Limit czasu: **3 s**

<Mistrzostwa w Algorytmice i Programowaniu>
FUNDACJA ROZWOJU INFORMATYKI
Ministerstwo Cyfryzacji
Uniwersytet Wrocławski

Krasnal Ekonomek jest biznesmenem. Ma on duży zbiór monet, które wymienia na giełdzie. Nie chce on Ciebie zanudzać szczegółami działania giełdy, więc po prostu zatrudnił Ciebie, aby móc śledzić, jak jego zbiór monet się zmienia w czasie.

Zbiór monet jest utożsamiany z ciągiem $A_1, A_2, \ldots, A_N$. Ekonomek wykonuje Q operacji, z czego niektóre to operacje na giełdzie, a pozostałe to zapytania o wartości monet, kierowane właśnie do Ciebie. Dokładniej, operacje te są następującej postaci:

- $1\ P\ V$ – podmień P-tą monetę na monetę o wartości V, co jest tożsame z wykonaniem $A_P = V$.
- $2\ S\ E\ V$ – podmień każdą monetę o wartości z przedziału $[S, E]$ na monetę o wartości V. Jest to tożsame z wykonaniem $A_P = V$, dla każdego P spełniającego $S \leqslant A_P \leqslant E$.
- $3\ P$ – wypisz wartość P-tej monety, czyli A_P.

Twoim zadaniem jest odpowiedzenie na jego zapytania.

Wejście

W pierwszym wierszu standardowego wejścia znajduje się jedna liczba naturalna N ($1 \leqslant N \leqslant 5 \cdot 10^5$) oznaczająca wielkość zbioru monet.

W drugim wierszu standardowego wejścia znajduje się ciąg N liczb całkowitych dodatnich $A_1, A_2, \ldots, A_N$ ($1 \leqslant A_i \leqslant 10^9$), określających wartości kolejnych monet w zbiorze Ekonomka.

W trzecim wierszu znajduje się jedna liczba całkowita Q $(1 \leqslant Q \leqslant 5 \cdot 10^5)$ oznaczająca liczbę zapytań. W kolejnych Q wierszach znajdują się opisy zapytań w formacie opisanym w treści zadania ($1 \leqslant P \leqslant N$, $1 \leqslant S \leqslant E \leqslant 10^9$, $1 \leqslant V \leqslant 10^9$).

Wyjście

Wypisz odpowiedź na każde zapytanie typu 3 w osobnym wierszu, w takiej kolejności w jakiej zapytania pojawiają się na wejściu. Możesz założyć, że pojawi się co najmniej jedno takie zapytanie.

Przykłady

Wejście

```
2
1 2
5
3 1
1 1 5
2 1 3 4
3 1
3 2
```

Wyjście

```
1
5
4
```

Wejście

```
5
1 2 3 4 5
12
2 2 4 5
2 1 1 10
3 1
3 2
3 5
1 2 7
2 7 10 1
3 1
3 2
3 3
3 4
3 5
```

Wyjście

```
10
5
5
1
1
5
5
5
```

Farmacjusz i jego roztwory

Zadanie: **F**
Limit pamięci: **256 MB**
Limit czasu: **4 s**

Krasnal Farmacjusz postanowił zrobić w końcu porządek w kolekcji swoich roztworów. Posiada on N probówek, które postanowił powkładać do pudełek. Jednakże sprawa okazała się trudniejsza niż przypuszczał...

Farmacjusz posiadał na początku jeden roztwór. Wszystkie pozostałe zostały uzyskane poprzez wszelakiego rodzaju obróbki jednego z wcześniej posiadanych. Krasnal nie pamięta już który roztwór był tym pierwotnym, jednakże jest w stanie na podstawie badań organoleptycznych określić które z nich są ze sobą spokrewnione. Innymi słowy, Farmacjusz jest w stanie wskazać takie pary roztworów, że jeden z nich powstał z drugiego (tylko nie wiadomo który z którego), a odtworzony przez niego graf pokrewieństwa roztworów jest acykliczny, nieskierowany i spójny.

Problem Farmacjusza polega na tym, że w momencie gdy włoży do jednego pudełka dwie próbówki ze spokrewnionymi roztworami, ich opary mogą się połączyć w niebezpieczną mieszankę. Ale nie jest to jedyna trudność – ze względu na cięcia w budżecie i aktualne promocje, wszystkie pudełka, które kupi Farmacjusz, muszą mieć taki sam rozmiar. Te dwie trudności sprawiły, że zupełnie się pogubił. Informacja jaka jest mu potrzebna, to ile pudełek musiałby kupić, gdyby wszystkie z nich były w jednakowym i z góry określonym rozmiarze? Czy jesteś w stanie mu pomóc?

Wejście

W pierwszym wierszu wejścia podana jest jedna liczba całkowita N $(1 \leqslant N \leqslant 10^6)$. Oznacza ona liczbę roztworów które posiada Farmacjusz.

W kolejnych $N-1$ wierszach znajdują się opisy spokrewnionych ze sobą roztworów. Każdy z nich składa się z dwóch oddzielonych pojedynczym odstępem liczb całkowitych a i b $(1 \leqslant a, b \leqslant N)$, oznaczających że roztwory a i b nie mogą trafić do jednego pudełka. Opis ten spełnia warunki z treści zadania.

Wyjście

W pierwszym i jedynym wierszu wyjścia wypisz dokładnie N liczb całkowitych, gdzie i-ta z nich oznacza minimalną liczbę pudełek o rozmiarze i potrzebnych do bezpiecznego zapakowania roztworów.

Przykład

Wejście

```
4
1 4
2 4
3 4
```

Wyjście

```
4 3 2 2
```

Gwiezdne Krasnale

Zadanie: **G**
Limit pamięci: **256 MB**
Limit czasu: **7 s**

Udało się! Krasnale podbijają kosmos! Wieloletnie prace nad własnym projektem kosmicznym doprowadziły do stworzenia statku kosmicznego MAŁY-3, który wyniósł załogę na orbitę okołoziemską. Niestety na tym sukcesy się skończyły. Gdy załoga obrała kurs na Marsa i gwałtownie przyspieszyła, doszło do awarii i MAŁY-3 stracił kontakt z bazą kontroli lotów.

MAŁY-3 przeniósł się do M-wymiarowej przestrzeni. Przyjmijmy taki układ współrzędnych, że statek znajduje się w jego środku (w punkcie o wszystkich współrzędnych równych 0). W kierunku statku zmierzają asteroidy, więc trzeba się natychmiast ewakuować z obecnej pozycji. Statek zostanie przez nie zniszczony, jeżeli będzie znajdował się w punkcie $(x_1, x_2, x_3 \ldots x_M)$ takim, że dla każdego $1 \leqslant i \leqslant M$, $x_i < B_i$. Kapitan obliczył już wszystkie wartości B_i.

Ze względu na awarię, załoga ma do dyspozycji jedynie N zapasowych silników. Każdy z nich może być w jednym z M rodzajów. Niestety ich działanie nie jest zbyt przewidywalne: zakładając, że statek znajduje się w punkcie $(x_1, x_2, x_3 \ldots x_M)$, po użyciu silnika k-tego rodzaju z prawdopodobieństwem $\frac{1}{2}$ może znaleźć się w punkcie $(x_1, x_2, \ldots x_k + 1, \ldots x_M)$ lub w $(x_1, x_2, \ldots x_k - 1, \ldots x_M)$ (również z prawdopodobieństwem $\frac{1}{2}$). Dostępne silniki mają jeszcze jedną poważną wadę, po użyciu psują się, więc każdego silnika można użyć co najwyżej raz.

Ponieważ na statku panuje panika, krasnale będą uruchamiać silniki w losowej kolejności. Jednak zachowały na tyle przytomności umysłu, że gdy opuszczą strefę przelotu asteroid, zatrzymają statek i przeczekają zagrożenie. Dla każdego k od 1 do N oblicz prawdopodobieństwo, że po uruchomieniu co najwyżej k silników będą bezpieczne, tzn. znajdą się w punkcie $(x_1, x_2, x_3 \ldots x_M)$, takim że $x_i \geqslant B_i$ dla pewnego $1 \leqslant i \leqslant M$.

Wynik należy wypisać modulo 998244353. Można pokazać, że szukane prawdopodobieństwo będzie liczbą wymierną, niech będzie równe p. Należy wypisać resztę z dzielenia $A \cdot C^{-1}$ przez 998244353, gdzie $p = \frac{A}{C}$, A i C są względnie pierwsze, C jest względnie pierwsze z 998244353, a C^{-1} jest multiplikatywną odwrotnością C modulo 998244353.

Należy założyć, że kolejność użycia silników jest losową permutacją (wybraną w taki sposób, że wszystkie permutacje są równie prawdopodobne) oraz kierunki działania poszczególnych silników są losowane niezależnie.

Wejście

W pierwszym wierszu wejścia znajdują się dwie liczby całkowite N i M ($1 \leqslant M, N \leqslant 5 \cdot 10^4$) oznaczające odpowiednio liczbę silników oraz wymiar. W drugim wierszu znajduje się N liczb całkowitych $k_1, k_2, \ldots k_N$ ($1 \leqslant k_i \leqslant M$). k_i oznacza rodzaj i-tego silnika. Trzeci wiersz zawiera M liczb naturalnych $B_1, B_2, \ldots B_M$ ($1 \leqslant B_i \leqslant N$) oznaczających granice strefy zagrożenia.

Wyjście

W jedynym wierszu wyjścia należy wypisać N liczb całkowitych. i-ta z nich jest równa prawdopodobieństwu (obliczonemu modulo 998244353), że krasnale uruchamiając co najwyżej i silników z punktu $(0, 0, \ldots 0)$ dotrą do takiego punktu $(x_1, x_2, \ldots x_M)$, że $x_j \geqslant B_j$ dla pewnej współrzędnej j.

Przykłady

Wejście

```
3 2
2 1 2
1 2
```

Wyjście

```
166374059 915057324 374341633
```

Aby opuścić strefę zagrożenia po uruchomieniu jednego silnika, krasnale muszą użyć drugiego silnika (jedynego typu 1), wtedy uda im się uciec do bezpiecznej strefy z prawdopodobieństwem $\frac{1}{2}$. Jedyny silnik typu 1 jest użyty jako pierwszy w dwóch spośród 6 kolejności używania silników (2, 1, 3 oraz 2, 3, 1). Wobec tego prawdopodobieństwo opuszczenia strefy zagrożenia w co najwyżej jednym ruchu jest równe $\frac{2}{6} \cdot \frac{1}{2} = \frac{1}{6}$.
Aby opuścić strefę zagrożenia w co najwyżej dwóch ruchach, krasnale mogą użyć dwóch silników typu 2 (z prawdopodobieństwem $\frac{2}{6}$) oraz obydwa te silniki muszą ich przesunąć w kierunku dodatnim (co zachodzi z prawdopodobieństwem $\frac{1}{2} \cdot \frac{1}{2}$). Inną możliwością jest użycie silnika typu 1 jako pierwszego lub drugiego w kolejności (z prawdopodobieństwem $\frac{4}{6}$). Wtedy krasnale muszą liczyć na to, że silnik zadziała w kierunku dodatnim (co zajdzie z prawdopodobieństwem $\frac{1}{2}$). Wobec tego, uruchamiając co najwyżej dwa silniki, krasnale ocalą statek z prawdopodobieństwem $\frac{2}{6} \cdot \frac{1}{4} + \frac{4}{6} \cdot \frac{1}{2} = \frac{5}{12}$
Aby krasnale opuściły strefę zagrożenia uruchamiając co najwyżej trzy silniki wystarczy, aby obydwa silniki typu 2 lub silnik typu 1 przesunęły statek w kierunku dodatnim. Te wydarzenia są niezależne. Pierwsze zachodzi w prawdopodobieństwem $\frac{1}{4}$, a drugie $\frac{1}{2}$. Wobec tego prawdopodobieństwo opuszczenia strefy niebezpieczeństwa przy użyciu co najwyżej trzech silników to $\frac{1}{4} + \frac{1}{2} - \frac{1}{4} \cdot \frac{1}{2} = \frac{5}{8}$.
Wobec tego kolejne wyniki (obliczone na liczbach wymiernych) są równe: $\frac{1}{6}$, $\frac{5}{12}$, $\frac{5}{8}$.

Wejście

```
4 3
2 2 2 1
2 1 1
```

Wyjście

```
623902721 499122177 467927041 374341633
```

W drugim teście kolejne wyniki (obliczone na liczbach wymiernych) są równe: $\frac{3}{8}$, $\frac{1}{2}$, $\frac{17}{32}$, $\frac{5}{8}$.

Hierotikon

Zadanie: **H**
Limit pamięci: **256 MB**
Limit czasu: **1 s**

<Mistrzostwa w Algorytmice i Programowaniu>

FUNDACJA ROZWOJU INFORMATYKI

Ministerstwo Cyfryzacji

Uniwersytet Wrocławski

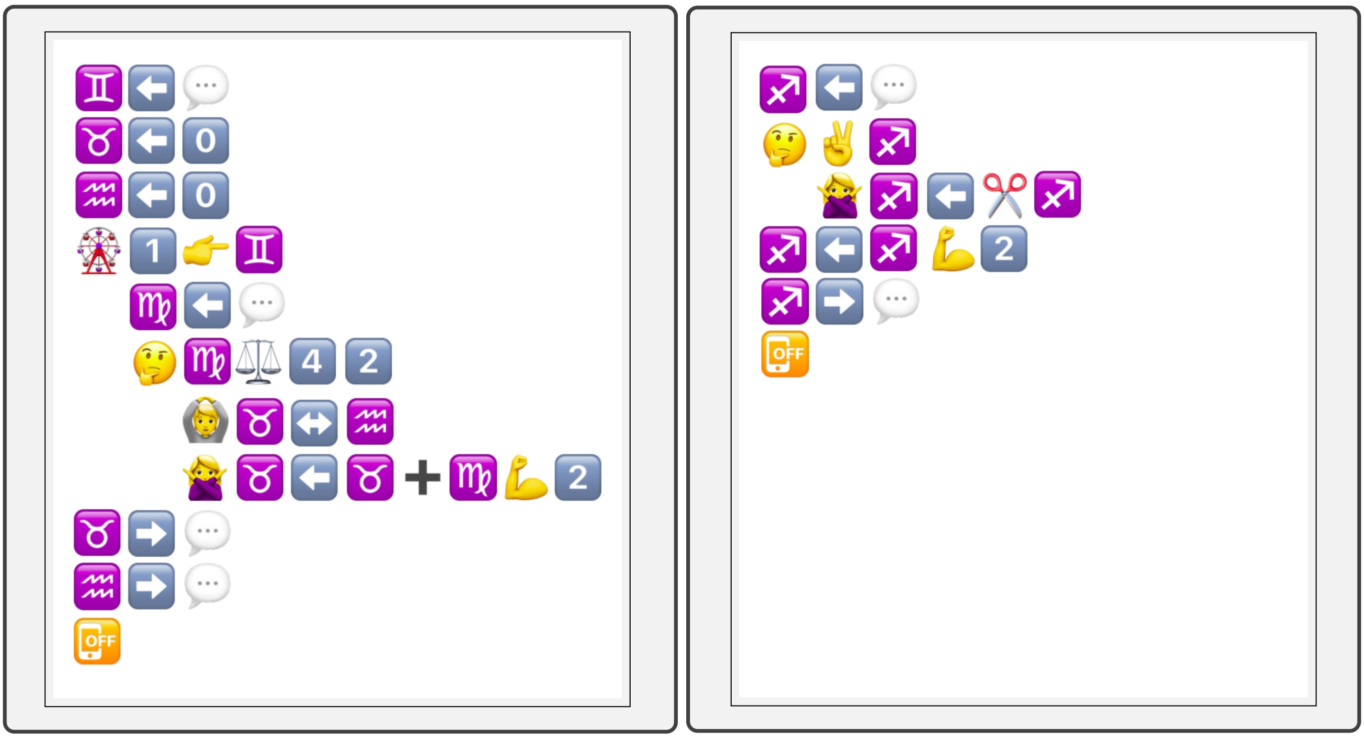
Krasnoludek Hieronim zawsze chciał sprawdzić jak poradziłby sobie w branży IT. Nigdy jednak nie mógł podjąć decyzji od jakiego języka programowania chciałby rozpocząć swoją przygodę z informatyką. Co odwlekało jego decyzję...

W dzisiejszych czasach tworzenie programów komputerowych jest znacznie łatwiejsze niż kilkadziesiąt lat temu. Najnowsze języki programowania starają się być jednocześnie przyjemne w obsłudze, jak i uniwersalne. Mimo wielu ułatwień względem ich poprzedników, nauka programowania jest jednak wciąż trudna... aż do dzisiaj! Przedstawiamy Wam język programowania Hierotikon™, który jest tak prosty i intuicyjny, że nie trzeba się go nawet uczyć.

Hieronim przeczytał dwa przykładowe programy, które umieściliśmy poniżej i od razu je zrozumiał! Postanowił rzucić Wam małe wyzwanie. Napisał on w tym celu pewien program w języku Hierotikon. Teraz pyta się Ciebie czy umiesz zaimplementować go w dowolnym języku akceptowanym na zawodach?

Po podaniu własnych danych możesz uruchomić programy w języku Hierotikon pod linkiem https://mistrzostwa.solve.edu.pl/zadanie_prog/.

Napisz program, który jest kopią programu Hieronima, a zatem dla poprawnych danych wejściowych obliczy i wypisze na standardowe wyjście to samo co program Hieronima.



Programy przykładowe.

Program do zaimplementowania.

Wejście

Wejście składa się z zera, jednej lub wielu liczb całkowitych oddzielonych pojedynczymi znakami odstępu. Wejście jest poprawne – jeżeli powyższy program w pewnym momencie spróbuje wczytać dane, to gwarantujemy, że znajdują się one na wejściu.

Wyjście

Wyjście składa się z zera, jednej lub wielu liczb całkowitych oddzielonych pojedynczymi znakami odstępu.

Ograniczenia

Wszystkie liczby na wejściu będą z przedziału $[1, 2 \cdot 10^5]$.

Przykład

Brak przykładów.

Wszystkie ikony użyte przy konstrukcji obrazków pozyskano ze strony: https://emojipedia.org/apple/

Intrygująca wyprawa

Zadanie: **I**
Limit pamięci: **256 MB**
Limit czasu: **2 s**

Krasnal Alpinek wybrał się na wycieczkę w góry. Tym razem jego trasa wiodła wzdłuż całej długości szlaku, który przechodził kolejno przez N szczytów. Na każdym z nich znajdował się punkt widokowy, a na nim tablica informacyjna dotycząca szczytu.

Niestety, podczas wyprawy Alpinka okazało się, że ktoś złośliwie zamazał wysokości wszystkich szczytów na tablicach, zastępując je wymyślonym przez siebie *współczynnikiem atrakcyjności szczytu*. Współczynnik ten oznaczał sumę wysokości trzech szczytów: szczytu na którym znajdowała się tabliczka, najwyższego szczytu na szlaku w kierunku jego początku oraz najwyższego szczytu na szlaku w kierunku jego końca.

Alpinek postanowił poszukać informacji na temat wysokości konkretnych szczytów u mieszkańców leżącej nieopodal wioski. Jednakże jedyną rzeczą, o której się dowiedział, było to, który ze szczytów jest najwyższy oraz jaka jest jego wysokość. Upewnił się on też, że najwyższy szczyt jest tylko jeden, więc wszystkie pozostałe szczyty są od niego ściśle niższe.

Alpinek próbuje teraz dociec na podstawie zebranych informacji jakie są wysokości kolejnych szczytów. Czy jesteś w stanie mu pomóc?

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita N $(1 \leqslant N \leqslant 10^6)$, oznaczająca liczbę szczytów na trasie Alpinka.

W drugim wierszu wejścia znajduje się N oddzielonych pojedynczymi odstępami liczb całkowitych s_i $(1 \leqslant s_i \leqslant 3 \cdot 10^8)$, gdzie s_i oznacza współczynnik atrakcyjności i-tego w kolejności szczytu. Formalniej, gdyby wysokości kolejnych szczytów wynosiły kolejno $a_1, a_2, \ldots, a_N$, to zachodzi $s_i = \max\limits_{1 \leqslant j \leqslant i-1}\{a_j\} + a_i + \max\limits_{i+1 \leqslant j \leqslant N}\{a_j\}$.

Możesz założyć, że wysokość każdego szczytu jest dodatnią liczbą całkowitą nie większą niż 10^8. Przyjmujemy umownie, że wysokości najwyższych szczytów na szlaku przed pierwszym szczytem oraz za ostatnim szczytem wynoszą 0.

W trzecim i ostatnim wierszu wejścia znajdują się dwie oddzielone pojedynczym odstępem liczby całkowite k oraz a_k $(1 \leqslant k \leqslant N; 1 \leqslant a_k \leqslant 10^8)$, oznaczające kolejno pozycję oraz wysokość najwyższego szczytu na trasie.

Wyjście

Na wyjściu wypisz N liczb $a_1, a_2, \ldots, a_N$ oznaczających wysokości kolejnych szczytów, spełniających wszystkie warunki podane w treści oraz na wejściu. Możesz założyć, że dla ciągu podanego na wejściu istnieje dokładnie jedna taka odpowiedź.

Przykłady

Wejście	Wyjście
5 9 10 14 13 8 4 6	3 1 5 6 2
4 12 15 19 16 4 10	2 3 6 10

Jeżynki

Zadanie: **J**
Limit pamięci: **256 MB**
Limit czasu: **2 s**

<Mistrzostwa w Algorytmice i Programowaniu>

FUNDACJA ROZWOJU INFORMATYKI

Ministerstwo Cyfryzacji

Uniwersytet Wrocławski

Krasnoludek Jędruś wybrał się na przechadzkę po swoim ogrodzie. W trakcie tej wyprawy zebrał pewną liczbę jeżyn oraz malin, po czym ustawił je w rzędzie o długości N. Owoce mają stać się poczęstunkiem dla znajomych Jędrusia. Niepokoi go jednak brak *wykwintności* ułożenia owoców. Uważa, że ciąg owoców jest wykwintny wtedy i tylko wtedy, gdy żadne dwa owoce tego samego gatunku nie sąsiadują ze sobą bezpośrednio.

Na szczęście Jędruś jest łakomczuchem i postanowił uczynić ciąg, który miał przed sobą, ciągiem wykwintnym, jednocześnie podjadając nadmierne owoce. Jędruś może wykonywać następującą operację:

- Wybierz dowolną parę sąsiadujących owoców, które są tego samego gatunku i usuń je z ciągu (od tej pory owoce na lewo i prawo od usuniętej pary stają się sąsiednie).

Jędruś nie jest miłośnikiem malin, ale wręcz przepada za jeżynami. Chciałby dowiedzieć się jaka jest największa liczba par jeżyn, które może usunąć z ciągu, tak aby stał się on wykwintny. Ponieważ do spotkania ze znajomymi pozostało jeszcze trochę czasu, to Jędruś postanowił urozmaicić nieco to zadanie. Chciałby rozważyć sumarycznie Q możliwości początkowych ustawień owoców. W każdym kolejnym zapytaniu będzie on zamieniał owoc na pewnej pozycji poprzedniego ustawienia na przeciwny (malinę na jeżynę, i na odwrót). Jego początkowy ciąg owoców oznaczymy jako ciąg binarny S_0, gdzie jedynki reprezentują jeżyny, a zera maliny. $f(X)$ to maksymalna liczba par jeżyn, które Jędruś może usunąć z ciągu X.

Do określenia, na których pozycjach Jędruś dokona zmian, potrzebny będzie ciąg p, który zdefiniowany jest następującym równaniem rekurencyjnym (gdzie A, B i M są stałymi podanymi na wejściu)

$$\begin{cases} p_0 = 0 \\ p_{i+1} = (A \cdot p_i + B) \bmod M \end{cases}$$

Ciąg S_{i+1} powstaje po zamianie owocu na $((p_i \bmod N) + 1)$-szej pozycji w ciągu S_i.

Napisz program, który obliczy $\sum_{i \in [1,Q]} f(S_i)$ i wypisze resztę z dzielenia wyniku przez $10^9 + 7$.

Wejście

W pierwszym wierszu wejścia znajdują się liczby N $(1 \leqslant N \leqslant 10^6)$ i Q $(1 \leqslant Q \leqslant 10^7)$, oznaczające długość słowa S_0 oraz liczbę rozważanych ciągów owoców. W drugim wierszu wejścia znajduje się słowo S_0. W trzecim wierszu wejścia znajdują się parametry A, B oraz M $(0 \leqslant A, B < M \leqslant 10^9 + 7)$.

Wyjście

W pierwszym wierszu wyjścia należy wypisać resztę z dzielenia żądanej sumy przez liczbę $10^9 + 7$.

Przykład

Wejście

```
5 5
11111
32 42 52
```

Wyjście

```
4
```

Pierwsze wyrazy ciągu p to $0, 42, 34, 38, 10$, więc zamieniamy owoce kolejno na pozycjach $1, 3, 5, 4, 1$.

Kolejne ułożenia owoców w teście przykładowym (zmiany podkreślone):

$\underline{0}1111$
$01\underline{0}11$
$0101\underline{0}$
$010\underline{0}0$
$\underline{1}1000$

Wyniki wywołań funkcji f na powyższych ciągach to $2, 1, 0, 0, 1$, więc odpowiedź to 4.

Karnety

Zadanie: **K**
Limit pamięci: **256 MB**
Limit czasu: **3 s**

<Mistrzostwa w Algorytmice i Programowaniu>

FUNDACJA ROZWOJU INFORMATYKI

Ministerstwo Cyfryzacji

Uniwersytet Wrocławski

Krasnal Karol wybrał się z rodziną na narty w Góry Krzesłowe. Po opłaceniu zakwaterowania w hotelu zaczął układać plan na najbliższe Q dni.

Puch i Sanki oraz *Pochyły Orczyk* to jedyne firmy, do których należą wyciągi w Górach Krzesłowych. Sieć wyciągów możemy przedstawić jako graf nieskierowany, w którym wierzchołki reprezentują wspólne dla firm stacje narciarskie, a krawędzie – wyciągi, z których każdy należy do jednej z firm. W momencie przyjazdu Karola wszystkie wyciągi są nieczynne.

Każdego dnia zachodzi jedno z dwóch zdarzeń:

- Między stacjami A_i i B_i zostaje uruchomiony na stałe nowy wyciąg należący do jednej z firm. Tego dnia stacje są zamknięte i Karol odpoczywa z rodziną w hotelu.
- Stacje są otwarte, więc Karol wybiera się na stok. Tego dnia planuje zostawić swój samochód na parkingu przy stacji A_i i za pomocą sieci wyciągów przemieścić się do stacji B_i.

Z obliczeń Karola wynika, że najkorzystniejszą opcją będzie wykupienie karnetów w obu firmach, które każdego dnia umożliwiają jednorazowe skorzystanie z wyciągów danej firmy. Rozumiemy przez to, że dozwolone jest przemieszczanie się wyciągami należącymi do jednej firmy oraz maksymalnie jedna zmiana przewoźnika, po której należy dostać się do docelowej stacji korzytając z wyciągów drugiej firmy.

Napisz program, który wczyta ze standardowego wejścia opis pobytu i dla każdego dnia, w którym stacje są otwarte, wypisze czy Karol jest w stanie przemieścić się do docelowej stacji.

Wejście

W pierwszym wierszu wejścia znajdują się dwie liczby naturalne N i Q ($1 \leqslant N, Q \leqslant 300\,000$), oznaczające odpowiednio liczbę stacji narciarskich oraz długość pobytu. Następnie Q linii opisuje kolejne dni pobytu w formacie:

- $1\ A_i\ B_i$ – firma *Puch i Sanki* uruchamia wyciąg między stacjami A_i i B_i,
- $2\ A_i\ B_i$ – firma *Pochyły Orczyk* uruchamia wyciąg między stacjami A_i i B_i,
- $3\ A_i\ B_i$ – Karol parkuje przy stacji A_i i chce przemieścić się do stacji B_i.

Wyjście

Dla każdego dnia, w którym Karol wybiera się na narty należy udzielić odpowiedzi w osobnej linii, wypisując `TAK`, jeśli Karol jest w stanie przemieścić się do docelowej stacji lub `NIE` w przeciwnym wypadku.

Przykłady

Wejście

```
4 8
3 1 1
1 1 2
2 2 3
1 3 4
3 1 3
3 1 4
1 2 3
3 1 4
```

Wyjście

```
TAK
TAK
NIE
TAK
```

Wejście	**Wyjście**
`4 7`	`NIE`
`1 1 1`	`TAK`
`2 2 3`	`TAK`
`3 1 2`	
`1 1 2`	
`3 1 2`	
`2 3 4`	
`3 1 4`	

Lody

Zadanie: **L**
Limit pamięci: **256 MB**
Limit czasu: **0.5 s**

<Mistrzostwa w Algorytmice i Programowaniu>

Krasnal Piastek uwielbia lody, jak i SMSowanie ze swoimi niskimi przyjaciółmi. Niestety, nie wziął on pod uwagę, że tych dwóch rzeczy nie należy robić jednocześnie. Chwila nieuwagi i BUM, nastąpiła katastrofa; cały telefon umazany w lodach! Na szczęście telefon wyszedł z tego niemal bez szwanku, poza jedną drobną usterką...

Został uszkodzony wskaźnik baterii w telefonie Piastka. Wskaźnik ten wyświetla dokładnie dwie cyfry, nawet jeśli poziom baterii jest poniżej 10%, przykładowo jeśli poziom baterii wynosi 7%, to wyświetla on 07. W wyniku wypadku z lodami, wskaźnik ten uległ niezwykle nietypowej awarii. Mianowicie w chwili, gdy wskazuje on $N\%$ baterii (oczywiście $N > 0$), to prawdziwy procent naładowania baterii to największa liczba $M < N$, taka że M różni się od N na dokładnie jednej cyfrze. Warto zauważyć, że dla $N > 0$, jest to dobrze zdefiniowane.

Krasnala Piastka przerosło obliczenie, ile procent baterii posiada obecnie jego telefon. Z tego względu zwrócił się do Ciebie z tym problemem. Twoim zadaniem jest odpowiedzenie na jego pytania.

Wejście

W pierwszym i jedynym wierszu standardowego wejścia znajduje się jedna liczba naturalna N ($1 \leqslant N \leqslant 99$) oznaczająca wskaźnik baterii wyświetlany w telefonie (bez zer wiodących).

Wyjście

W pierwszym i jedynym wierszu standardowego wyjścia powinna znaleźć się odpowiedź na pytanie Piastka – prawdziwy procent naładowania baterii.

Przykłady

Wejście
79

Wyjście
78

Liczba 78 jest największą liczbą mniejszą od 79, która różni się od niej na dokładnie jednej cyfrze (cyfrze jedności). Chociaż liczba 77 również różni się na tylko jednej pozycji od 79, to liczba 78 jest poprawną odpowiedzią, ponieważ chcemy znaleźć największą taką liczbę.

Wejście
80

Wyjście
70

Liczba 70 jest największą mniejszą od 80, która różni się od niej na dokładnie jednej cyfrze (cyfrze dziesiątek).

Wejście
7

Wyjście
6

Wejście
1

Wyjście
0

Męczące podchody

Zadanie: **M**
Limit pamięci: **256 MB**
Limit czasu: **1 s**

<Mistrzostwa w Algorytmice i Programowaniu>

FUNDACJA ROZWOJU INFORMATYKI

Ministerstwo Cyfryzacji

Uniwersytet Wrocławski

Krasnal Leszko bawi się ze swoimi przyjaciółmi w podchody w lesie. Na szczęście las zna on jak własną kieszeń – składa się on z N polan połączonych ścieżkami. Akurat tak się składa, że z każdej polany da się dojść ścieżkami do każdej innej, ale tylko na jeden sposób (nie przechodząc przez żadną polanę więcej niż jednokrotnie).

Zabawa w podchody jest bardzo prosta – na początku wszyscy zaczynają na jednej polanie (oznaczmy ją numerem 1). Następnie przyjaciele Leszka wybierają sobie pewną polanę, po czym idą na nią najkrótszą możliwą trasą, aby schować się tam razem. Zadaniem Leszka jest jak najszybsze znalezienie polany, na której są znajomi. Aby trochę uprościć Leszkowi zadanie, jego przyjaciele na każdej polanie po której przeszli po drodze zostawiają na ziemi znak *X*, aby wiedział on, że idzie w dobrym kierunku. Jednakże będąc na polanie Leszko musi sam zdecydować w którą stronę dalej szukać.

Czy jesteś w stanie policzyć jaki jest najmniejszy czas poszukiwań, po którym Leszko może być pewny, że znalazł swoich znajomych, niezależnie od tego na której polanie się schowali?

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita N $(1 \leqslant N \leqslant 3 \cdot 10^5)$ oznaczająca liczbę polan w lesie. Polany ponumerowane są kolejnymi liczbami całkowitymi od 1 do N, gdzie 1 oznacza numer polany, na której rozpoczyna się zabawa.

W kolejnych $N-1$ wierszach znajdują się opisy pojedynczych ścieżek. Każdy z nich składa się z dwóch oddzielonych pojedynczym odstępem liczb a i b $(1 \leqslant a, b \leqslant N)$, oznaczających że pomiędzy polanami a i b znajduje się ścieżka. Przejście każdej ścieżki zajmuje dokładnie 1 minutę, za to przekroczenie polany jest na tyle krótkie, że możemy je pominąć.

Wyjście

W pierwszym i jedynym wierszu wyjścia wypisz jedną liczbę całkowitą oznaczającą najmniejszy możliwy czas poszukiwań, po jakim Leszko może być pewny, że znajdzie swoich znajomych, niezależnie od tego gdzie się schowali.

Przykłady

Wejście	**Wyjście**
4 1 2 1 3 1 4	5

Wejście	Wyjście
10 1 2 2 3 3 4 3 5 6 1 6 7 6 8 8 9 7 10	7

Nieustraszony akrobata

Zadanie: **N**
Limit pamięci: **256 MB**
Limit czasu: **1 s**

Podczas tegorocznego festynu czeka nas jeszcze jedna nie lada atrakcja – występ akrobatyczny nieustraszonego krasnala Latałka. Specjalnie na tę okazję przygotowanych zostało N trampolin, a każda z nich umieszczona jest na pewnej wysokości. Organizatorzy nie przewidzieli jednak jednej rzeczy – pomimo, że Latałek nie boi się niczego, to technicznie jest on w stanie przeskoczyć z jednej trampoliny na drugą tylko, jeśli wartość bezwzględna różnicy ich wysokości wynosi co najwyżej K (ale trampoliny te nie muszą sąsiadować ze sobą).

Wiadomo już, że występ Latałka rozpocznie się na trampolinie o numerze 1, po czym wykona on pewną sekwencję skoków pomiędzy trampolinami. Czy jesteś w stanie policzyć na ilu różnych trampolinach może się on zakończyć?

Wejście

W pierwszym wierszu standardowego wejścia znajdują się dwie liczby naturalne N $(1 \leqslant N \leqslant 5 \cdot 10^5)$ oraz K $(1 \leqslant K \leqslant 10^9)$ oznaczające odpowiednio liczbę trampolin przygotowanych do występu oraz zasięg skoku Latałka. W drugim wierszu standardowego wejścia znajduje się N liczb całkowitych $a_1, a_2, \ldots, a_N$ $(1 \leqslant a_i \leqslant 10^9)$ oznaczających wysokości kolejnych trampolin. Latałek rozpocznie swój występ na trampolinie o wysokości a_1.

Wyjście

W jedynym wierszu standardowego wyjścia powinna znaleźć się jedna liczba całkowita będąca liczbą trampolin, na które jest w stanie dotrzeć Latałek, wykonując pewną sekwencję skoków.

Przykłady

Wejście	Wyjście
5 2 3 1 8 4 5	4
7 1 4 2 4 8 8 4 5	4

Ogrodnik

<Mistrzostwa w Algorytmice i Programowaniu>
FUNDACJA ROZWOJU INFORMATYKI
Ministerstwo Cyfryzacji
Uniwersytet Wrocławski

Krasnalowi Ogrodnikowi znudziło się już doglądanie róż i tulipanów. Stwierdził, że hodowla rosiczek będzie znacznie bardziej ekscytująca. Planuje je karmić mrówkami. Przygotował już prostopadłościenne terrarium o wymiarach podstawy $A \times B$, w którym będzie trzymał owady.

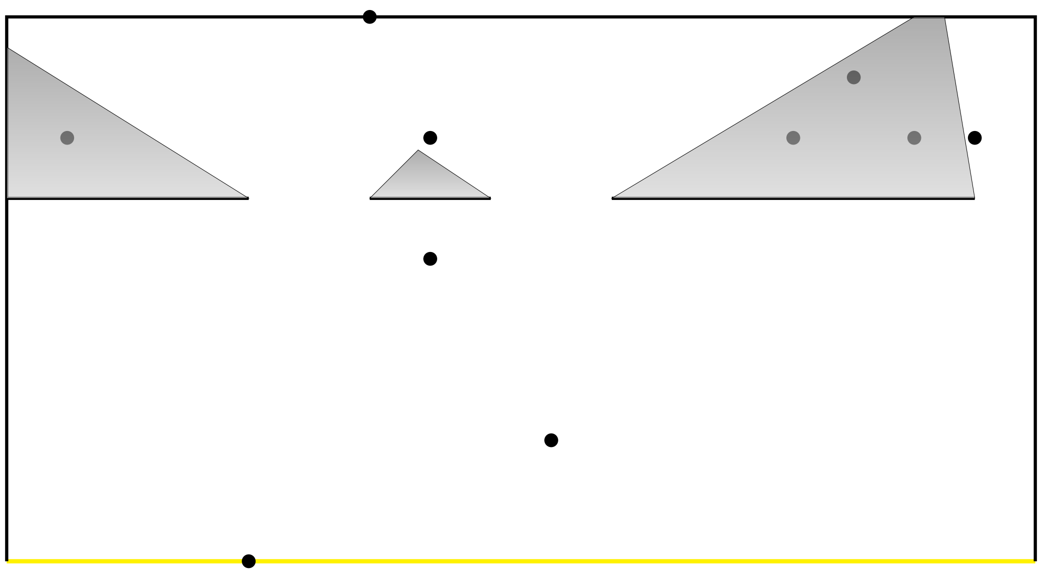
Niestety, rosiczki jeszcze nie zdążyły wykiełkować, więc naszemu małemu bohaterowi nudzi się i ogląda N mrówek biegających po terrarium. Aby trochę urozmaicić sobie i mrówkom życie, Ogrodnik postanowił założyć w terrarium oświetlenie i umieścić w nim M przeszkód.

Aby dokładnie opisać strukturę terrarium przyjmiemy układ współrzędnych o osiach prostopadłych do boków podstawy. Przyjmiemy, że lewy dolny róg ma współrzędne $(0, 0)$, natomiast prawy górny (A, B). Oświetlenie będzie się składać ze świetlówki umieszczonej na odcinku $[(0, 0), (A, 0)]$.

Przeszkody to cienkie nieprzeźroczyste płytki ustawione na sztorc równolegle do osi OX. Przeszkody nie dotykają ani nie przecinają się. Ponadto znajdują się w równej odległości od świetlówki, więc można je reprezentować jako rozłączne odcinki na prostej $y = K$, dla pewnej stałej wartości K.

Krasnal właśnie miał zgasić światło w pokoju, aby sprawdzić, jak działa oświetlenie terrarium. Wie, że mrówek jest dokładnie N, jednak ile z nich będzie widocznych? Znając pozycje mrówek, wyznacz ile z nich znajduje się poza cieniem rzucanym przez przeszkody.

Mrówki są na tyle małe, że powinny być traktowane jako punkty. Zakładamy też, że mrówki nie rzucają cienia.



Wejście

W pierwszym wierszu wejścia znajduje się pięć liczb naturalnych N, M, A, B, K ($1 \leqslant N, M \leqslant 500000$, $2 \leqslant A, B \leqslant 10^9$, $1 \leqslant K \leqslant B$) oznaczające odpowiednio liczbę mrówek, przeszkód, wymiary terrarium oraz współrzędną y prostej, na której leżą przeszkody.

Następne M wierszy zawiera opisy przeszkód. Każdy z nich zawiera dwie liczby całkowite x_1, x_2 $(0 \leqslant x_1 < x_2 \leqslant A)$ oznaczające dekorację o końcach $(x_1, K), (x_2, K)$.

Kolejne N wierszy zawiera po dwie liczby całkowite x, y $(0 \leqslant x \leqslant A$, $0 \leqslant y \leqslant B)$ oznaczające, że pewna mrówka znajduje się w punkcie (x, y).

Możesz założyć, że odcinki oraz punkty podane na wejściu będą rozłączne. Ponadto żadna z mrówek nie będzie znajdowała się na granicy cienia.

Wyjście

W jedynym wierszu wyjścia należy wypisać liczbę mrówek, które nie znajdą się w cieniu.

Przykład

Wejście

```
10 3 17 9 6
0 4
6 8
10 16
4 0
9 2
7 5
7 7
6 9
1 7
13 7
14 8
15 7
16 7
```

Wyjście

```
6
```

Sytuacja z tego testu jest przedstawiona na rysunku. Świetlówka znajduje się na dolnym boku (zaznaczona na żółto), zaznaczone punkty oznaczają pozycje mrówek, a zaciemnione obszary cień rzucany przez przeszkody.