

Trening (A)

Limit pamięci: 64 MB

Limit czasu: 2.00 s

Jasio postanowił, że wystartuje w Internetowych Turniejach Programistycznych. W ramach ambitnych przygotowań wybrał już N zadań, które planuje rozwiązać w ramach treningu. Niestety, jak to Jasio, zaczął trochę późno, a do zawodów zostało już niewiele czasu. Każdego dnia jest w stanie rozwiązać co najwyżej K zadań, zanim opadnie z sił. Pomóż Jasiowi oszacować minimalną liczbę dni, które musi poświęcić, aby rozwiązać wszystkie zadania.

Wejście

W pierwszym (jedynym) wierszu wejścia znajdują się dwie liczby naturalne N i K , oddzielone pojedynczym odstępem.

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć jedna liczba naturalna, oznaczająca minimalną liczbę dni, które Jasio musi poświęcić, aby rozwiązać wszystkie N problemów.

Ograniczenia

$$1 \leq N, K \leq 10^9$$

Przykład

Wejście

9 3

Wyjście

3

Wyjaśnienie

Jasio może każdego dnia rozwiązywać po 3 zadania. Wówczas po 3 dniach rozwiąże wszystkie 9.

Wejście

5 4

Wyjście

2

Wyjaśnienie

Jasio musi poświęcić co najmniej 2 dni na rozwiązanie wybranych 5 problemów, ponieważ jednego dnia jest on w stanie rozwiązać co najwyżej 4 z nich.

Andrzej Hood (B)

Limit pamięci: 128 MB

Limit czasu: 5.00 s

Andrzej Hood z Bytewood jest bohaterem znanym w całej Bajtocji. Żyje on wyłącznie po to, by redystrybuować dobra przekazując je z rąk ludzi bogatych do rąk ludzi biednych. Jak już wspomniałem, zamieszkuje on las Bytewood, który składa się z polanek i drózek leśnych między nimi. Z każdej polanki a można dojść do każdej polanki b na dokładnie jeden sposób.¹

Na każdej dróżce czai się banda Andrzeja Hooda, gotowa, by redystrybuować. Oto jak działa jej system:

- Każda dróżka ma przypisany zbiór towarów, na jakie dana banda poluje (towary będą oznaczane liczbami naturalnymi od 1 do 64).
- Jeśli podróżny przechodzący przez dróżkę posiada towar należący do przypisanego jej zbioru, to zostaje on zabrany przez bandę.
- Jeśli podróżny nie posiada towaru należącego do przypisanego jej zbioru, banda wręcza mu go w prezencie.²

Bogaci kupcy zauważyli, że w tym systemie istnieją szczególne, bezpieczne szlaki,³ które finalnie nie zmieniają ekwipunku podróżnych, niezależnie od tego, z czym wyruszają. Innymi słowy, na takich szlakach, bandy zabierają każdy towar dokładnie tyle samo razy, ile razy go dają.

Twoim zadaniem jest napisać program, który wyznaczy liczbę wszystkich bezpiecznych szlaków.

Dwa szlaki uznajemy za różne, jeśli istnieje przynajmniej jedna dróżka, która wchodzi w skład pierwszego i nie wchodzi w skład drugiego. W szczególności, dla dowolnych a, b , szlak łączący polany o numerach a i b jest tożsamy ze szlakiem łączącym polany o numerach b i a .

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N oznaczająca liczbę polanek.

W kolejnych $N - 1$ wierszach wejścia znajduje się opis drózek.

Na początku i -tego z nich znajdują się trzy liczby naturalne $1 \leq a_i, b_i \leq N, 0 \leq k_i \leq 64$, oddzielone pojedynczymi odstępami. Oznaczają one, że istnieje dróżka łącząca polanki o numerach a_i i b_i , do której przypisany jest zbiór mający k_i elementów.

Po nich znajduje się k_i **różnych**, dodatnich liczb naturalnych, nie większych niż 64, pooddzielanych pojedynczymi odstępami. Liczby te są elementami zbioru przypisanego do i -tej dróżki.

Wyjście

W pierwszym i jedynym wierszu wyjścia należy wypisać jedną liczbę całkowitą, oznaczającą liczbę bezpiecznych szlaków.

Ograniczenia

$$2 \leq N \leq 500\,000$$

$$0 \leq k_i \leq 64$$

Przykład

Wejście

```
6
1 2 1 2
2 3 1 1
2 4 0
2 5 1 1
1 6 2 1 2
```

Wyjście

```
4
```

Wyjaśnienie

W wersji HTML poniżej znajduje się grafika obrazująca test przykładowy. Jedyne bezpieczne szlaki łączą polanki o numerach: (3, 6), (2, 4), (3, 5), oraz (5, 6).

¹Właściwie to Bytewood jest drzewem, ale ludzie nie żyją na drzewach.

²Wszystkie bandy mają nieskończone ilości każdego towaru i mogą go bezkarnie rozdawać.

³Szlak to po prostu ścieżka, nie mylić z dróżką.

Łączony WF (c)

Limit pamięci: 64 MB

Limit czasu: 2.00 s

W szkole Jasia jest N klas. Kilka z tych klas ma odbyć wspólną lekcję wuefu, na której będą grały w piłkę. Żeby gra była uczciwa, wszystkich uczestników należy podzielić na dwie drużyny składające się z takiej samej liczby osób. Z drugiej strony, żeby gra była fajna, sumarycznie uczestników powinno być jak najwięcej.

Oczywiście nie można podzielić klasy na części – albo cała klasa bierze udział w lekcji albo nikt z danej klasy nie bierze w niej udziału. Może się jednak zdarzyć, że dwie osoby z tej samej klasy zostaną przydzielone do różnych drużyn.

Napisz program, który wczyta liczbę klas oraz liczebność każdej z nich, a następnie wypisze, ile osób może wziąć udział w łączonym wuefie przy najlepszym możliwym wyborze klas.

Wejście

W pierwszym wierszu wejścia znajduje się jedna dodatnia liczba naturalna N oznaczająca liczbę klas. W drugim wierszu wejścia znajduje się N dodatnich liczb naturalnych, podzielanych pojedynczymi odstępami. Oznaczają one liczebności kolejnych klas.

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć jedna liczba całkowita będąca największą możliwą liczbą uczestników łączonego wuefu.

Ograniczenia

$$1 \leq N \leq 1\,000\,000$$

Każda klasa ma nie więcej niż 10^9 uczniów.

Przykład

Wejście

```
4
2 4 3 2
```

Wyjście

```
8
```

Wyjaśnienie

Najlepszy możliwy wybór to klasy 1, 2 i 4.

Wejście

```
1
5
```

Wyjście

```
0
```

Wyjaśnienie

Niestety, żaden wybór klas nie pozwala na zorganizowanie wuefu.

Wejście

```
5
2 6 3 2 1
```

Wyjście

```
14
```

Wyjaśnienie

Szczęśliwie wszyscy mogą wziąć udział w łączonym wuefie.

Porządek musi być! (D)

Limit pamięci: 256 MB

Limit czasu: 2.00 s

Jasio czuje się już całkiem pewnie w porównywaniu słów leksykograficznie. W końcu to nic trudnego – wystarczy czytać litery od lewej do prawej, aż trafi się na pierwszą różnicę, która rozstrzyga, które słowo jest „mniejsze”. Zainspirowany tą prostą metodą, Jasio chwycił za kartkę i zaczął porównywać słowa na własną rękę... ale szybko zrobił się z tego niezły bałagan. Teraz potrzebuje Twojej pomocy, zanim wszystko się poplącze jeszcze bardziej!

Na początku Jasio ma jedno puste słowo. Następnie wykonuje N zapytań, z których każde jest jednym z trzech poniższych typów:

- 1 a b – Jasio dopisuje literę b na koniec słowa o numerze a.
- 2 a – Jasio tworzy nowe słowo, będące kopią słowa o numerze a.
- 3 a b – Jasio prosi Cię o porównanie leksykograficznie słów o numerach a i b.

Słowa są numerowane w kolejności ich tworzenia. Początkowe słowo ma numer 1, kolejnym słowom przypisujemy numery 2, 3, 4, ...

Czy dasz radę poprawnie odpowiedzieć na wszystkie zapytania Jasia?

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N , oznaczająca liczbę zapytań. W kolejnych N wierszach znajduje się opis kolejnych zapytań w formacie takim, jak przedstawiono w treści.

Wyjście

Na wyjściu należy wypisać tyle wierszy, ile było zapytań typu 3. W każdym z tych wierszy powinna znaleźć się odpowiedź na odpowiednie zapytanie, zgodnie z kolejnością podaną na wejściu. Odpowiedź powinna spełniać poniższy format:

- < – w przypadku, gdy pierwsze słowo jest mniejsze leksykograficznie od drugiego.
- = – w przypadku, gdy słowa się nie różnią.
- > – w przypadku, gdy drugie słowo jest mniejsze leksykograficznie od pierwszego.

Ograniczenia

$1 \leq N \leq 200\,000$, Jasio dopisuje do słów tylko małe litery alfabetu angielskiego.

Przykład

Wejście

```
6
1 1 j
3 1 1
1 1 a
2 1
1 2 n
3 1 2
```

Wyjście

```
=
<
```

Wyjaśnienie

Po kolejnych operacjach lista słów wygląda następująco:

- 1 1 j: Dopisujemy literę j do pierwszego słowa.
- 3 1 1: Porównujemy pierwsze słowo ze sobą, $j = j$.
- 1 1 a: Dopisujemy literę a do pierwszego słowa, teraz pierwsze słowo to ja.
- 2 1: Tworzymy drugie słowo ja.
- 1 2 n: Dopisujemy literę n do drugiego słowa.
- 3 1 2: Porównujemy pierwsze słowo z drugim. $ja < jan$.

Wejście

7
1 1 a
1 1 b
2 1
3 1 2
1 1 c
1 2 d
3 2 1

Wyjście

=
>

Ranking Jasia (E)

Limit pamięci: 256 MB

Limit czasu: 5.00 s

Jasio zapisał się na Internetowy Turniej Programistyczny, w którym bierze udział N zawodników. Z podekscytowaniem oczekuje na rywalizację i chciałby już teraz dowiedzieć się, które miejsce może zająć. Na szczęście Jasio zna się na ludziach i potrafi porównać umiejętności niektórych uczestników między sobą. W szczególności wyznaczył już M par (a_i, b_i) , co do których jest przekonany, że zawodnik o numerze a_i uzyska wyższe miejsce (miejsce o mniejszym numerze) w turnieju od zawodnika o numerze b_i .

Twoim zadaniem jest napisanie programu, który na podstawie podanych przez Jasia informacji określi najwyższe oraz najniższe miejsce, jakie może on zająć w Turnieju.

Jasio oczywiście mógł się pomylić i podać sprzeczne informacje,¹ co również powinno zostać wykryte przez Twój program.

Wejście

W pierwszym wierszu wejścia znajdują się dwie liczby naturalne N i M , oznaczające odpowiednio liczbę zawodników oraz liczbę informacji podanych przez Jasia.

W i -tym z kolejnych M wierszy znajdują się dwie **różne** liczby naturalne a_i i b_i , oznaczające, że zawodnik o numerze a_i uzyska lepszy wynik od zawodnika o numerze b_i .

Uczestnicy są ponumerowani liczbami naturalnymi od 1 do N , a Jasio jest zawodnikiem o numerze 1.

Zagwarantowane jest, że wszystkie **uporządkowane** pary (a_i, b_i) są parami różne.

Wyjście

Program powinien wypisać dwie liczby naturalne, oddzielone spacją, oznaczające odpowiednio najwyższe oraz najniższe miejsce, na którym może znaleźć się Jasio. W przypadku sprzecznych informacji, zamiast tego należy wypisać jedno słowo NIE.

Ograniczenia

$1 \leq N \leq 1\,000\,000$, $0 \leq M \leq 1\,000\,000$

Przykład

Wejście

4 4
2 4
1 3
2 1
4 3

Wyjście

2 3

Wyjaśnienie

Zawodnik numer 2 uzyska wyższe miejsce niż Jasio(1), a zawodnik nr 3 niższe. O zawodniku nr 4 nie wiadomo, czy będzie wyżej, czy niżej od Jasia w rankingu. Zatem najwyższe miejsce, które może zająć Jasio to 2, a najniższe 3.

Wejście

5 4
2 1
3 2
1 3
4 5

Wyjście

NIE

Wyjaśnienie

Jasio informacje są sprzeczne, ponieważ wynika z nich, że zawodnik 2 jest lepszy od Jasia, zawodnik 3 lepszy od zawodnika 2, a Jasio lepszy od zawodnika 3.

Wejście

Wyjście

Wyjaśnienie

¹Czyli takie, że dowolne przypisanie uczestnikom miejsc nie spełnia przynajmniej jednej z nich.

3 2
1 2
2 3

1 1

Jasio ma zapewnione 1 miejsce,
ponieważ jest lepszy od zawodnika 2,
który jest lepszy od zawodnika 3.

Wielka szachownica (F)

Limit pamięci: 64 MB

Limit czasu: 2.00 s

Był zimowy, wigilijny wieczór. W domu Jasia pachniała choinka, a pod jej gałęziami piętrzyły się kolorowe prezenty. Wśród nich znajdowała się jedna, spora paczka z podpisem *dla Jasia*. *Playbox* — pomyślał Jasio i z podekscytowaniem rozpakował prezent. W środku odkrył dużą, piękną szachownicę o wymiarach $N \times N$.

Od tamtej chwili minęło już wiele lat, a czarna farba pokrywająca co drugie pole uległa znacznemu zatarciu. Jasio chciałby odnowić swoją szachownicę. Nie pamięta jednak, które pola były czarne, a które białe. Wie tylko tyle, że pole o współrzędnych $(1, 1)$ było **czarne**. Twoim zadaniem będzie napisać program, który wczyta współrzędne jednego pola szachownicy i wypisze, czy jest ono czarne, czy białe.

Wejście

W pierwszym (jedynym) wierszu wejścia znajdują się trzy liczby całkowite N , x , y oddzielone pojedynczymi odstępami, oznaczające kolejno: rozmiar szachownicy oraz numer kolumny i numer rzędu pola, o które pytamy.

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinien znaleźć się napis CZARNE albo BIALE, zgodny z kolorem wejściowego pola.

Ograniczenia

$$1 \leq N \leq 10^{18}$$

$$1 \leq x, y \leq N$$

Przykład

Wejście

10 5 6

Wyjście

BIALE

Wyjaśnienie

W wersji HTML poniżej znajduje się rysunek przedstawiający szachownicę o wymiarach 10×10 z zaznaczonym polem z testu przykładowego.

Tuje (G)

Limit pamięci: 256 MB

Limit czasu: 1.00 s

Bitold jest znany na swoim osiedlu jako właściciel wyjątkowo zadbanego ogrodu. Jego dumą i radością jest elegancki rząd tui, które tworzą gęsty, równiutki żywopłot. Najważniejsze dla Bitolda jest, aby każda tuja miała identyczną wysokość, co nadaje ogrodowi harmonijny wygląd. Niestety, sąsiad Bitolda, Karol, pełen zazdrości, pewnej nocy zakradł się do ogrodu i przyciął niektóre tuje, czyniąc je nierównymi i zdeformowanymi. Bitold nie zamierzał się poddać. W ramach zemsty i z nieugiętą determinacją postanowił jak najszybciej przywrócić swoim tujom jednakową wysokość. Sięgnął po tajną broń – nawóz firmy ByteGarden, który jest znany ze swojej niezwyklej skuteczności. Rozsypanie garści nawozu pod daną tują sprawia, że jej wysokość zwiększa się o 2, a wysokość sąsiadujących z nią tui zwiększa się o 1. Nie chcąc marnować nawozu, Bitold będzie go rozsypywał tylko pod tujami sąsiadującymi z dwoma innymi (nie będącymi na początku lub końcu rzędu). Twoim zadaniem jest pomóc Bitoldowi przywrócić równą wysokość wszystkich tui, zużywając jak najmniej nawozu. Opracuj plan, który wskaże, ile garści nawozu powinien rozsypać pod każdą z tui, aby żywopłot znów był idealnie równy i ogród odzyskał swój dawny blask. Spraw, by ogród Bitolda znów stał się najpiękniejszym miejscem na osiedlu, a zazdrość Karola obróciła się przeciwko niemu.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N , oznaczająca liczbę tui tworzących żywopłot. W drugim wierszu wejścia znajduje się ciąg N liczb naturalnych T_i , pooddzielanych pojedynczymi odstępami, gdzie T_i określa wysokość i -tej tui w rzędzie, po przycięciu.

Wyjście

W jedynym wierszu wyjścia należy wypisać ciąg N liczb naturalnych A_i , gdzie A_i oznacza liczbę garści nawozu, które należy rozsypać pod i -tą tują. Ciąg powinien zaczynać się i kończyć zerem. Jeżeli rozwiązanie nie istnieje, zamiast tego należy wypisać tylko jedno słowo NIE.

Ograniczenia

$5 \leq N \leq 1\,000\,000$, $0 \leq T_i \leq 1\,000\,000$.

Przykład

Wejście

```
6
2 0 1 2 3 4
```

Wyjście

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

Wyjaśnienie

W wersji HTML poniżej znajduje się grafika obrazująca test przykładowy. Początkowe wysokości tui 2 0 1 2 3 4
Aplikacja nawozu +2 +1 Końcowe, wyrównane wysokości tui 4 4 4 4 4 4

Wejście

```
5
7 5 6 3 5
```

Wyjście

```
NIE
```

Cukierki (H)

Limit pamięci: 32 MB

Limit czasu: 1.00 s

Jasio kupił N cukierków, ponieważ uwielbia je podjadać podczas rozwiązywania zadań. Jako iż cukier bywa uzależniający, każdego dnia będzie jadł o jednego cukierka więcej niż dnia poprzedniego. Teraz Jasio zastanawia się, na ile sposobów może ustalić liczbę cukierków, które zje pierwszego dnia, tak aby każdego kolejnego dnia mógł zwiększać liczbę jedzonych cukierków o dokładnie 1, aż do momentu zjedzenia wszystkich N cukierków. Napisz program, który wczyta liczbę cukierków i wypisze wynik na standardowe wyjście.

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się jedna liczba naturalna N , oznaczająca liczbę kupionych przez Jasia cukierków.

Wyjście

W pierwszym (jedynym) wierszu wyjścia należy wypisać jedną liczbę naturalną – liczbę sposobów, na które Jasio może ustalić liczbę cukierków, które zje pierwszego dnia.

Ograniczenia

$$1 \leq N \leq 10^{13}$$

Przykład

Wejście

5

Wyjście

2

Wyjaśnienie

Jasio może zjeść 5 cukierków na raz albo pierwszego dnia zjeść 2 cukierki i drugiego 3.

Wejście

18

Wyjście

3

Wyjaśnienie

Jasio może rozpocząć jedzenie od 3 ($3 + 4 + 5 + 6 = 18$), 5 ($5 + 6 + 7 = 18$) lub 18 cukierków.

Wejście

1

Wyjście

2

Wyjaśnienie

Jasio może zacząć od 0 ($0 + 1 = 1$) lub 1 cukierka.

Trójkąty równoramienne (I)

Limit pamięci: 64 MB

Limit czasu: 5.00 s

Na płaszczyźnie znajduje się N **parami różnych** punktów. Twoim zadaniem jest napisać program, który wczyta ich współrzędne, a następnie wyznaczy i wypisze liczbę różnych, trzelementowych podzbiorów¹ zbioru wszystkich N punktów, będących zbiorami wierzchołków jakiegoś niezdegenerowanego² trójkąta **równoramiennego**.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba całkowita N , oznaczająca liczbę wszystkich punktów. W i -tym z kolejnych N wierszy znajdują się dwie liczby całkowite x_i, y_i , oddzielone pojedynczym odstępem, oznaczające współrzędne i -tego punktu.

Wyjście

W pierwszym (i jedynym) wierszu wyjścia powinna się znaleźć jedna liczba całkowita będąca odpowiedzią do zadania.

Ograniczenia

$$1 \leq N \leq 3000$$

$$-10^9 \leq x_i, y_i \leq 10^9$$

Przykład

Wejście

```
5
0 0
-1 -1
-1 1
1 -1
1 1
```

Wyjście

```
8
```

Wyjaśnienie

W wersji HTML poniżej znajduje się rysunek

Szukane podzbiory to $\{1, 2, 3\}$, $\{1, 3, 5\}$, $\{1, 4, 5\}$, $\{1, 2, 4\}$, $\{2, 3, 5\}$, $\{2, 3, 4\}$, $\{2, 4, 5\}$, $\{3, 4, 5\}$.

¹Czyli trójki bez porządku. Przykładowo: $\{1, 2, 3\}$ i $\{2, 3, 1\}$ to ten sam zbiór i powinien być policzony co najwyżej raz.

²Zbiór wierzchołków niezdegenerowanego trójkąta definiujemy jako zbiór trzech niewspółliniowych punktów. Dodatkowo, trójkąt (zbiór punktów) ten jest równoramienny, gdy w trójkącie (figurze geometrycznej), powstałym przez dorysowanie odcinków między punktami, istnieją co najmniej dwa ramiona, które są równe.