

Kamień, papier, nożyce (A)

Limit pamięci: 32 MB

Limit czasu: 0.50 s

Jasio gra z Małgosią w kamień, papier, nożyce. Jest to gra, w której dwaj gracze równocześnie pokazują sobie symbole reprezentujące odpowiednio: kamień, papier lub nożyce. Jeżeli gracze pokazali sobie te same symbole – jest wtedy remis. W przeciwnym przypadku: kamień wygrywa z nożycami (tępi je), nożyce wygrywają z papierem (tną go), a papier wygrywa z kamieniem (owija go). Zasady te są nieco zbyt skomplikowane dla Jasia. Pomóż mu ustalić, kto wygrał grę.

Wejście

W pierwszym i drugim wierszu wejścia znajduje się jedno ze słów KAMIEN, PAPIER, NOZYCE (pisane wielkimi literami, bez polskich znaków), określające symbole pokazane przez Jasia (w pierwszym wierszu) oraz Małgosię (w drugim wierszu).

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinno się znaleźć jedno ze słów JASIO, MALGOSIA lub REMIS (pisane wielkimi literami, bez polskich znaków), określające imię zwycięzcy rozgrywki opisanej na wejściu (REMIS w przypadku remisu).

Przykład

Wejście

KAMIEN

PAPIER

Wyjście

MALGOSIA

Suma cyfr sumy cyfr (B)

Limit pamięci: 32 MB

Limit czasu: 0.50 s

Funkcja $s(n)$ dla liczby naturalnej n zwraca jej sumę cyfr. Przykładowo: $s(123) = 1 + 2 + 3 = 6$.
Dana jest liczba naturalna N . Wskaż najmniejszą liczbę naturalną x , dla której $s(s(x)) = N$.

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się jedna liczba naturalna N .

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się jedna liczba naturalna x .

Ograniczenia

$1 \leq N \leq 60$.

Przykład

Wejście

10

Wyjście

199

Liczba nawiasowań (c)

Limit pamięci: 32 MB

Limit czasu: 2.00 s

Zbiór poprawnych nawiasowań można zdefiniować rekurencyjnie:

1. pusty napis jest poprawnym nawiasowaniem,
2. jeżeli X jest poprawnym nawiasowaniem to: (X) również jest poprawnym nawiasowaniem,
3. jeżeli P oraz Q są poprawnymi nawiasowaniami to PQ (sklejenie P i Q ze sobą) też jest poprawnym nawiasowaniem,
4. wszystkie poprawne nawiasowania można wyprowadzić za pomocą powyższych reguł.

Łatwo zauważyć, że każde poprawne nawiasowanie musi mieć parzystą długość. Dla ustalonego N , w miarę nietrudno też policzyć ile jest poprawnych nawiasowań długości $2N$:

- dla $N = 1$ (nawiasowania długości 2) jest tylko jedno takie nawiasowanie: $()$,
- dla $N = 2$ (nawiasowania długości 4) są dwa takie nawiasowania: $(())$ oraz $()()$,
- dla $N = 3$ (nawiasowania długości 6) jest pięć poprawnych nawiasowań: $((()))$, $()(())$, $(())()$, $(())()$ oraz $()()()$.

W ogólności: liczba poprawnych nawiasowań o określonej długości jest ściśle związana z ciągiem liczb Catalana C_n . Jest dokładnie C_n poprawnych nawiasowań długości $2n$.

Wzorów na obliczenie n -tej liczby Catalana jest wiele. Przykładowo takie:

- $C_n = \frac{1}{n+1} \binom{2n}{n}$,
- $C_n = \frac{1}{2n+1} \binom{2n+1}{n}$,
- $C_n = \binom{2n}{n} - \binom{2n}{n+1}$,
- $C_0 = 1, C_n = \sum_{i=1}^n C_{i-1} C_{n-i}$ (dla $n > 0$),
- $C_0 = 1, C_n = \frac{4n-2}{n+1} C_{n-1}$ (dla $n > 0$).

W tym zadaniu rozszerzamy definicję poprawnych nawiasowań na większą liczbę typów nawiasów: w punkcie drugim definicji mówimy, że poprawne są również nawiasowania $[X]$ oraz $\{X\}$. Rozważamy więc poprawne (uogólnione) nawiasowania złożone z nawiasów $(,), \{, \}, [,]$. Czy potrafisz powiedzieć ile jest uogólnionych poprawnych nawiasowań o długości $2N$? Sprawdźmy to! Ponieważ wynik może być duży, wystarczy nam poznać jego resztę z dzielenia przez $10^9 + 7$.

Napisz program, który: wczyta liczbę N , wyznaczy resztę z dzielenia przez $10^9 + 7$ liczby uogólnionych poprawnych nawiasowań długości $2N$, i wypisze wynik na standardowe wyjście.

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się jedna liczba naturalna N .

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć reszta z dzielenia przez $10^9 + 7$.

Ograniczenia

$1 \leq N \leq 2000$.

Ciekawostka: Możliwe jest rozwiązanie tego zadania dla znacznie większych limitów (np. organizatorzy turnieju potrafią napisać program, który mieści się w limitach czasu nawet przy ograniczeniu $N \leq 10^{11}$). Ale to jest turniej dla początkujących. Zachęcamy do przemyślenia szczegółów po turnieju.

Przykład

Wejście

2

Wyjście

18

Wyjaśnienie

Chodzi przykładowo o nawiasowania

 $()()$, $\{[]\}$ lub $[()]$.

Oś liczbowa (D)

Limit pamięci: 32 MB

Limit czasu: 0.50 s

Bajtek napisał profesjonalny program do wyświetlania osi liczbowej w postaci ASCII artu (znaków - (minus), + (plus), > (znak większości), odstępów oraz cyfr).

Przykładowe osie liczbowe, które może wygenerować profesjonalny program Bajtka mogą wyglądać tak:

```
-----+-----+-----+----->      --+---+----->      +-----+>
   155           158           160           3330 3340           0           9
```

Jak widać na powyższych przykładach, efekt końcowy jest bardzo schludny. Formalizując jednak opis tych osi, Bajtek przyjął następujące założenia:

- każda oś liczbowa składa się z dwóch wierszy opisu o długości nie przekraczającej 1 000 000 znaków,
- pierwszy wiersz opisu składa się ze znaków - (fragment poziomy osi liczbowej), + (oznaczenie zaznaczonego punktu na osi) oraz > (na końcu osi liczbowej),
- etykiety zaznaczonych punktów na osi liczbowej są w drugim wierszu, pod znakami +, możliwie wyśrodkowane (w przypadku liczb parzystej długości dokładnie połowa cyfr znajduje się po lewej stronie znaku),
- oznaczenia punktów (znaki + oraz liczbowe etykiety poniżej) nie wystają poza oś liczbową ani z lewej strony, ani z prawej strony
- etykiety punktów są nieujemnymi liczbami całkowitymi i nie przekraczają 10^9 ,
- liczbowe etykiety punktów nie nakładają się na siebie i zawsze jest między nimi co najmniej jeden znak odstępu,
- na osi zaznaczone są zawsze co najmniej dwa punkty,
- punkty zaznaczone na osi są dokładnie (tzn. nie są stosowane żadne przybliżenia wynikające z „rozdzielczości” znaków).

Twoim zadaniem jest napisać program, który będzie zaznaczał dodatkowe punkty na osi wygenerowanej przez program Bajtka.

Wejście

W pierwszych dwóch wierszach znajduje się opis osi liczbowej, zgodnie z warunkami podanymi powyżej.

W trzecim wierszu znajduje się jedna liczba naturalna N oznaczająca liczbę dodatkowych punktów, które należy zaznaczyć na osi liczbowej. W czwartym (ostatnim) wierszu znajduje się ciąg N nieujemnych liczb całkowitych X_i pooddzielanych pojedynczymi odstępami. Są to współrzędne punktów, które należy dodatkowo zaznaczyć na osi liczbowej. Gwarantowane jest, że te punkty nie zostały zaznaczone na osi podanej w pierwszych dwóch wierszach wejścia.

Możesz założyć, że dane są dobrane w taki sposób, że możliwe jest zaznaczenie dodatkowych punktów, aby nadal spełniać założenia Bajtka opisane powyżej. Ponadto, zaznaczenie wszystkich dodatkowych punktów jest możliwe na podanej osi liczbowej, bez jej rozszerzania w którąkolwiek ze stron.

Wyjście

Twój program powinien wypisać oś liczbową z zaznaczonymi dodatkowymi punktami (ponad te, które były już wcześniej zaznaczone) w takim samym formacie jak pierwsze dwa wiersze wejścia.

Długość pierwszego wiersza wyjścia musi być taka sama jak długość pierwszego wiersza wejścia. Niedopuszczalne jest przesunięcie istniejących już punktów w inne miejsca.

Ograniczenia

$$1 \leq N \leq 100\,000, 0 \leq X_i \leq 10^9.$$

Przykład

Wejście

-----+-----+----->
155 158 160
2
161 157

Wyjście

-----+-----+-----+----->
155 157 158 160 161

Wejście

--+-----+----->
3330 3340
1
3364

Wyjście

--+-----+-----+----->
3330 3340 3364

Wejście

+-----+>
0 9
3
2 7 4

Wyjście

+--+--+--+>
0 2 4 7 9

Świat według Fibonacciego (E)

Limit pamięci: 32 MB

Limit czasu: 2.50 s

Zachowanie Jasia ostatnio zaczęło bardzo odbiegać od normy. Zafascynował się ciągiem Fibonacciego. Różne są jego definicje, ale w tym zadaniu przyjmujemy, że jest to ciąg rozpoczynający się od dwóch jedynek, w którym każdy kolejny element jest sumą dwóch poprzednich elementów (1, 1, 2, 3, 5, 8, 13, ...). W szczególności, zakładamy, że 0 **nie jest** liczbą Fibonacciego.

Samo zafascynowanie nie byłoby jeszcze czymś dziwnym, ale Bajtek widzi teraz liczby Fibonacciego dosłownie wszędzie. Przykładowo: w liczbie 12 nie ma prawdopodobnie zbyt wiele specjalnego, a już w szczególności w kontekście liczb Fibonacciego.

Bajtek powiedziałby jednak coś takiego: *Przecież jeśli wziąć resztę z dzielenia liczby 12 przez 7, to dostajemy 5, czyli liczbę Fibonacciego!* Zapewne prawie o każdej liczbie można by tak powiedzieć: wszystkie liczby nieparzyste dają resztę 1 (czyli liczbę Fibonacciego) przy dzieleniu przez 2, wszystkie liczby niepodzielne przez 3 dają reszty 1 lub 2 (czyli ponownie liczby Fibonacciego) przy dzieleniu przez 3, więc niewiele w tym specjalnego.

Bajtek dla każdej liczby X potrafi podać bardzo dużo przykładów liczb M , że reszta z dzielenia liczby X przez M jest liczbą Fibonacciego. A to już podejrzane. Może rzeczywiście jest tak, że każda liczba ma coś w sobie z liczby Fibonacciego? Czy możesz to sprawdzić?

Napisz program, który: wczyta liczbę X , wyznaczy liczbę wartości $M < X$, takich, że reszta z dzielenia X przez M jest liczbą Fibonacciego i wypisze wynik na standardowe wyjście.

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się jedna liczba naturalna X .

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć liczba różnych wartości $M < X$, że reszta z dzielenia X przez M jest liczbą Fibonacciego.

Ograniczenia

$$1 \leq X \leq 10^{12}.$$

Przykład

Wejście

12

Wyjście

5

Wyjaśnienie

W tym przypadku mamy $M \in \{5, 7, 9, 10, 11\}$.

Kąty wielokąta foremnego (F)

Limit pamięci: 32 MB

Limit czasu: 0.50 s

Wielokąt nazywamy foremnym, gdy wszystkie jego boki są równej długości oraz wszystkie jego kąty są równej miary. Przykładowo: trójkąt równoboczny jest wielokątem foremnym, podobnie jak kwadrat.

Jasio ma pewien wielokąt foremny. Chciałby wiedzieć ile boków ma jego wielokąt. Mógłby oczywiście policzyć, ale jest bardzo leniwy. Zauważył bowiem, że możliwe jest obliczenie liczby boków wielokąta foremnego na podstawie miary jednego z jego kątów wewnętrznych. Zmierzył więc miarę tego kąta i... na tyle wystarczyło mu siły. Czy pomożesz mu ustalić liczbę boków jego wielokąta?

Wejście

W pierwszym (jedynym) wierszu wejścia znajduje się jedna liczba naturalna α – miara kąta wewnętrznego w wielokącie foremnym, podana w stopniach.

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć jedna liczba naturalna – liczba boków wielokąta foremnego.

Jeżeli Jasio się pomylił i nie istnieje wielokąt foremny o takiej miarze kąta wewnętrznego, zamiast tego należy wypisać tylko jedno słowo NIE.

Ograniczenia

$$0 \leq \alpha \leq 360.$$

Przykład

Wejście

60

Wyjście

3

Wejście

90

Wyjście

4

Wejście

71

Wyjście

NIE

Zadanie dla dzieci (G)

Limit pamięci: 32 MB

Limit czasu: 0.50 s

Jasio znalazł w książce dla dzieci następujące zadanie:

Ile minimalnie ruchów potrzeba, aby z poniższego ustawienia uzyskać sytuację, w której na wierzchu są same reszki? W każdym ruchu należy obrócić na drugą stronę **dokładnie trzy** (niekoniecznie sąsiednie) monety.



Zadanie bardzo mu się spodobało, szybko je rozwiązał i zaczął się zastanawiać nad modyfikacjami tego zadania:

- A co gdyby monet nie było 5, tylko N ?
- A co gdyby na początku nie były cztery reszki, tylko R ?
- A co gdyby w każdym ruchu można było przewrócić na drugą stronę nie trzy, ale dokładnie K monet?

Pomóż Jasiowi i napisz program, który wczyta wartości N , R oraz K i ustali minimalną liczbę ruchów potrzebnych do osiągnięcia celu (ustawienia z samymi reszkami na wierzchu) lub ustali, że osiągnięcie celu jest niemożliwe i wypisze wynik na standardowe wyjście.

Wejście

W pierwszym (jedynym) wierszu wejścia znajdują się trzy nieujemne liczby całkowite N , R oraz K , podzielane pojedynczymi odstępami.

Wyjście

W pierwszym (jedynym) wierszu wyjścia powinna się znaleźć jedna liczba naturalna – minimalna liczba ruchów niezbędnych do osiągnięcia celu.

Jeżeli jest to niemożliwe, zamiast tego należy wypisać tylko jedno słowo NIE.

Ograniczenia

$1 \leq N \leq 5\,000$, $0 \leq R \leq N$, $0 \leq K \leq N$.

Ciekawostka: Możliwe jest rozwiązanie tego zadania dla znacznie większych limitów (na pewno da się powiększyć limit na N do 10^6 , a możliwe, że nawet bardziej). Ale to jest turniej dla początkujących. Zachęcamy do przemyślenia szczegółów po turnieju.

Przykład

Wejście

5 4 3

Wyjście

3