Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie
Ogólnopolska Olimpiada
"O Diamentowy Indeks AGH" 2020/21

Informatyka – Etap I

Zadanie 1 (15 punktów)

Dana jest macierz kwadratowa A o wymiarach $N \times N$. Na macierzy A możemy wykonać trzy typy operacji:

1. *Rotation*: Obróć macierz o kąt S (S jest zawsze wielokrotnością kąta $90°$).
2. *Query*: Zapytaj o wartość elementu macierzy o indeksach R i C.
3. *Update*: Zmiana wartości elementu macierzy o indeksach R i C na V.

Wejście

W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite N i M oznaczające odpowiednio rozmiar macierzy A oraz liczbę operacji. W kolejnych N wierszach znajduje się po dokładnie N liczb całkowitych, stanowiących kolejne wiersze macierzy A. Następne M wierszy zawierają opisy kolejnych operacji składające się z pojedynczego znaku (A, Q lub U) oznaczającego typ operacji i od 1 do 3 liczb całkowitych (zależnie od typu operacji).

1. `A S`: rotacja macierzy o kąt S
2. `Q R C`: pytanie o element o indeksach R, C
3. `U R C V`: zmiana wartości elementu o indeksach R, C na nową wartość V

Ograniczenia

- $1 \leqslant N \leqslant 1000$
- $1 \leqslant A_{ij} \leqslant 1000$
- $0 \leqslant R, C < N$
- S jest wielokrotnością kąta $90°$

Uwaga: Wszystkie operacje typu *Update* są wykonywane na **początkowej** macierzy. Po zmianie wartości wszystkie poprzednie rotacje muszą być wykonane na zaktualizowanej macierzy.

Wyjście

Standardowe wyjście powinno się składać z jednego wiersza dla **każdej** operacji typu Q (zapytanie). Każdy wiersz powinien zawierać dokładnie jedną liczbę całkowitą: wartość elementu macierzy A o indeksach R i C (zadanych w pytaniu) w jej aktualnej postaci (czyli po rotacjach i aktualizacjach zadanych poprzednimi operacjami).

Przykład

Dla danych wejściowych:

```
2 7
1 2
3 4
A 90
Q 0 0
Q 0 1
A 90
Q 0 0
U 0 0 6
Q 1 1
```

poprawną odpowiedzią jest:

```
3
1
4
6
```

Wyjaśnienie:

Początkowa macierz:

```
1 2
3 4
```

Rotacja: Po obrocie o 90°:

```
3 1
4 2
```

Pytanie: Element $A_{00} = 3$

Pytanie: Element $A_{01} = 1$

Rotacja: Po obrocie o 90°:

```
4 3
2 1
```

Pytanie: Element $A_{00} = 4$

Aktualizacja (początkowej macierzy):

```
6 2
3 4
```

Po aktualizacji stosujemy poprzednie obroty (czyli w sumie 180°)

```
4 3
2 6
```

Pytanie: Element $A_{11} = 6$

Zadanie 2 (15 punktów)

Mamy daną tablicę A dodatnich liczb całkowitych o długości N, na której możemy wykonać co najwyżej K operacji. Operacja jest zdefiniowana następująco:

1. Wybierz dowolny element tablicy A ($A[i]$)
2. Zastąp $A[i]$ przez `floor`$(A[i]/2)$

Proszę napisać program wyznaczający *minimalną* sumę elementów tablicy po wykonaniu na niej co najwyżej K operacji.

Wejście

W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite N i K oznaczające odpowiednio długość tablicy A i maksymalną liczbę operacji. Kolejny wiersz zawiera N liczb całkowitych: wartości tablicy.

Ograniczenia

- $1 \leqslant N,\ K \leqslant 10^6$
- $1 \leqslant A[i] \leqslant 10^9,\ i = 0, 1, \ldots, N-1$

Wyjście

Standardowe wyjście powinno zawierać jedną liczbę całkowitą: minimalną sumę elementów tablicy po wykonaniu co najwyżej K operacji.

Przykład

Dla danych wejściowych:

```
4 3
20 7 5 4
```

poprawną odpowiedzią jest:

```
17
```

Zadanie 3 (15 punktów)

Mamy daną tablicę A liczb całkowitych o długości N oraz liczbę całkowitą K. Element tablicy A_i ma parę, jeżeli w tablicy znajduje się inny element, $A_j \neq A_i$, o wartości z przedziału $[A_i - K,\ A_i + K]$

Proszę napisać program, który wylicza liczbę elementów, które mają parę.

Wejście

W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite N i K oznaczające odpowiednio długość tablicy A i rozmiar przedziału, jak opisano wyżej. Kolejny wiersz zawiera N liczb całkowitych: wartości tablicy A.

Ograniczenia

- $1 \leqslant N,\ K \leqslant 10^6$
- $0 \leqslant A_i \leqslant 10^9,\ i = 0,\ 1,\ \ldots,\ N-1$

Wyjście

Standardowe wyjście powinno zawierać jedną liczbę całkowitą: liczbę elementów tablicy A, które mają parę.

Przykład

Dla danych wejściowych:

```
7 3
5 5 7 9 15 2 15
```

poprawną odpowiedzią jest:

```
5
```

Wszystkie elementy poza 15 mają parę w przedziale $[A_i - 3,\ A_i + 3]$. Wprawdzie w tablicy są dwa elementy o wartości 15, ale nie stanowią one dla siebie pary, ponieważ są równe.

Zadanie 4 (15 punktów)

Mając daną dodatnią liczbę całkowitą N, stwórzmy nową liczbę dodając kwadraty cyfr liczby N. Można udowodnić, że postępując w ten sposób wielokrotnie otrzymamy w końcu wynik 1 lub 4.

Przykład:

$13 = 1^2 + 3^2 = 1 + 9 = 10$ (Krok 1)

$10 = 1^2 + 0^2 = 1 + 0 = 1$ (Krok 2, kończymy iterację ponieważ uzyskaliśmy liczbę 1)

Jeżeli w opisanej powyżej procedurze uzyskamy wynik 1, to liczbę N nazywamy "jedno-kwadratową".

Proszę napisać program, który znajduje K-tą liczbę w zadanym przedziale $[L, U]$, która jest jednocześnie jednokwadratowa i pierwsza.

Wejście

W pierwszym wierszu standardowego wejścia znajdują się trzy liczby całkowite L, U i K. L i U oznaczają odpowiednio dolną i górną granicę przedziału, w którym poszukujemy liczb (włączając L i U). K jest numerem liczby w tak uzyskanym ciągu.

Ograniczenia

- $1 \leqslant L \leqslant U \leqslant 10^9$
- $K \geqslant 1$

Wyjście

Standardowe wyjście powinno zawierać jedną liczbę całkowitą: K-tą liczbę jednokwadratową i pierwszą w zadanym przedziale. Jeżeli w przedziale nie ma co najmniej K takich liczb, program wypisuje wartość -1.

Przykłady

1. Dla danych wejściowych:

   ```
   1 30 3
   ```

   poprawną odpowiedzią jest:

   ```
   19
   ```

2. Dla danych wejściowych:

   ```
   12 33 5
   ```

   poprawną odpowiedzią jest:

   ```
   -1
   ```

Zadanie 5 (20 punktów)

Niech $p[]$ oznacza sekwencję liczb pierwszych w porządku rosnącym, czyli $p[0] = 2,\ p[1] = 3,\ p[2] = 5,\ \ldots$. Dane są dwie liczby całkowite, N i D. Dla każdego $0 \leqslant i < N$, niech $q[i] = p[i] \times p[i + D]$. Proszę napisać program wyznaczający liczbę rozwiązań równania $A + B + C + D = E$ takich, że $A \leqslant B \leqslant C \leqslant D$ i wszystkie liczby A, B, C, D, E należą do zbioru $\{q[0],\ q[1],\ \ldots,\ q[N - 1]\}$.

Wejście

W pierwszym i jedynym wierszu standardowego wejścia znajdują się dwie liczby całkowite N i D.

Ograniczenia

- $1 \leqslant N \leqslant 2500$,
- $0 \leqslant D \leqslant 2500$,

Wyjście

W pierwszym i jedynym wierszu standardowego wyjścia program powinien wypisać jedną liczbę całkowitą będącą rozwiązaniem.

Przykłady

1. Dla danych wejściowych:

   ```
   15 1
   ```

   poprawną odpowiedzią jest:

   ```
   2
   ```

   Rozwiązaniami równania spełniającymi założenia są:

   ```
   6 + 15 + 323 + 323 = 667
   6 + 143 + 221 + 1147 = 1517
   ```

   Uwaga: Pierwsze rozwiązanie używa wartości 323 dwukrotnie - nie jest to sprzeczne z warunkami zadania.

2. Dla danych wejściowych:

   ```
   2470 0
   ```

   poprawną odpowiedzią jest:

   ```
   0
   ```

   Nie istnieje rozwiązanie spełniające założenia.

Zadanie 6 (20 punktów)

Dana jest dwuwymiarowa tablica znakowa $M[R][C]$ zawierająca znaki '.' lub '#'. Znak '.' oznacza pusty element, znak '#' oznacza ścianę. Zakładamy, że labirynt jest otoczony ze wszystkich stron przez dodatkowe ściany, nie uwzględnione na wejściu programu.

Startujemy z elementu $M[0][0]$. Celem jest dotarcie do elementu $M[R-1][C-1]$. W każdym kroku możemy przemieścić o jedną pozycję w prawo, w lewo, w górę lub w dół, ale **nie wolno** wchodzić na ściany.

W zestawach podanych na wejściu programu dojście do celu jest niemożliwe. Jednak w **dokładnie jednym kroku** możliwe jest usunięcie jednej ściany przylegającej do bieżącej pozycji.

Proszę napisać program, który znajduje wszystkie takie ściany, że po usunięciu jednej z nich będzie możliwe dojście do celu $M[R-1][C-1]$. Program powinien zwrócić liczbę takich ścian. Program zwraca zero, jeżeli po usunięciu dowolnej ściany nie jest możliwe dotarcie do celu.

Wejście

W pierwszym wierszu standardowego wejścia znajdują się dwie liczby całkowite R i C zawierająca odpowiednio liczbę wierszy i kolumn tablicy M. W kolejnych R wierszach znajduje się po dokładnie C znaków, stanowiących kolejne wiersze tablicy.

Ograniczenia

- $2 \leqslant R \leqslant 100$
- $2 \leqslant C \leqslant 100$
- Każdy znak w tablicy będzie znakiem '.' lub '#'
- $M[0][0]$ i $M[R-1][C-1]$ będzie zawsze '.'
- Nie jest możliwe przejście z $M[0][0]$ do $M[R-1][C-1]$

Wyjście

W pierwszym i jedynym wierszu standardowego wyjścia program powinien wypisać jedną liczbę całkowitą będącą rozwiązaniem.

Przykłady

1. Dla danych wejściowych:

   ```
   3 3
   ..#
   .#.
   #..
   ```

   poprawną odpowiedzią jest:

   ```
   3
   ```

   Usunięcie dowolnej z trzech ścian umożliwia dojście do celu.

2. Dla danych wejściowych:

   ```
   4 6
   ..##..
   ..##..
   ...#..
   ..##..
   ```

   poprawną odpowiedzią jest:

   ```
   1
   ```