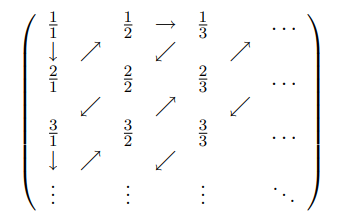
E. Нумерация дробей

|  |  |
| --- | --- |
| Ограничение времени | 2 секунды |
| Ограничение памяти | 256Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Георг Кантор доказал, что множество всех рациональных чисел счетно (т.е. все рациональные числа можно пронумеровать).

Чтобы упорядочить дроби необходимо их положить в таблицу, как показано на рисунке. В строку с номером *i* этой матрицы по порядку записаны дроби с числителем *i*, а в столбец с номером *j* дроби с знаменателем *j*.



Дальше необходимо выписать все дроби в том порядке, как показано на рисунке стрелками. Получится такая последовательность: , , , , , , *…*

Вам требуется по числу *n* найти числитель и знаменатель *n*-ой дроби.

Формат ввода

Во входном файле дано число *n* (*1 ≤ n ≤ 1018*) — порядковый номер дроби в последовательности.

Формат вывода

В выходной файл требуется вывести через символ / два числа: числитель и знаменатель соответствующей дроби.

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 1 | 1/1 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 6 | 3/1 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2 | 2/1 |

def GetLength(num: int) -> int:

return (num \* (1 + num)) // 2

n = int(input())

left = 0

right = n

while left < right:

mid = (left + right) // 2

if n <= GetLength (mid):

right = mid

else:

left = mid + 1

numberOfDiagonal = left

numberInDiagonal= GetLength(numberOfDiagonal) - n

numerator = 0

denominator = 0

if numberOfDiagonal % 2 == 1:

numerator = numberOfDiagonal - numberInDiagonal

denominator = numberInDiagonal + 1

else:

numerator = numberInDiagonal + 1

denominator = numberOfDiagonal - numberInDiagonal

print(f'{numerator}/{denominator}')