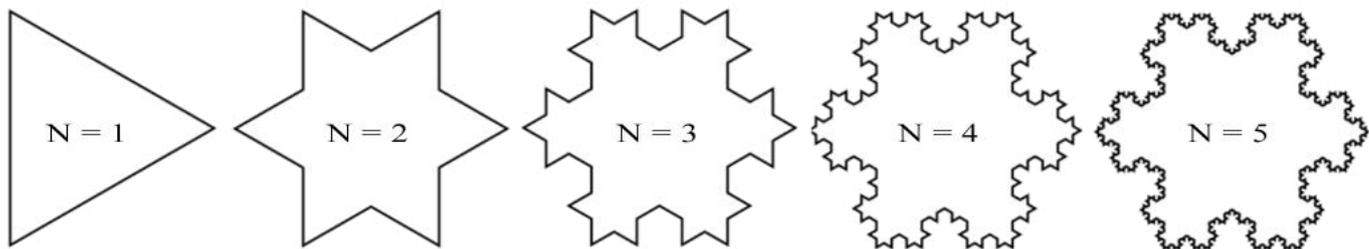


## Задача А. Снежинка Коха

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 Мб

Снежинка Коха – фрактальная кривая, которая строится на основе равностороннего треугольника, который представляет собой снежинку Коха первого порядка ( $N=1$ ). Снежинка Коха  $K$ -го порядка строится из подобной кривой  $(K-1)$ -го порядка ( $K > 1$ ) путем замены каждой стороны данной фигуры четырьмя отрезками, каждый из которых представляет  $1/3$  от длины исходного отрезка (см. рисунок).



По заданному значению  $N$  требуется определить площадь фигуры, ограниченной снежинкой Коха  $N$ -го порядка, полагая, что при  $N=1$  площадь равна единице.

### Входные данные

Входной файл INPUT.TXT содержит натуральное число  $N$  ( $N \leq 10^{18}$ ).

### Выходные данные

В выходной файл OUTPUT.TXT выведите площадь, ограниченную снежинкой Коха  $N$ -го порядка не менее чем с шестью знаками после десятичной точки.

### Примеры

| № | INPUT.TXT | OUTPUT.TXT   |
|---|-----------|--------------|
| 1 | 1         | 1.000000     |
| 2 | 2         | 1.3333333333 |

### Система оценивания

Решения, работающие только для  $N \leq 5$ , будут оцениваться в 40 баллов.

Решения, работающие только для  $N \leq 10^6$ , будут оцениваться в 90 баллов.

## Задача В. Голосование

*Ограничение по времени: 1 секунда*

*Ограничение по памяти: 16 Мб*

В выборах участвовало несколько кандидатов. В результате проведения тайного голосования был получен список из  $N$  фамилий. Каждая фамилия – это голос, отданный участником голосования в пользу этого кандидата.

На основании этих данных требуется построить гистограмму результатов проведенного голосования.

### Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число  $N$  – количество отданных голосов ( $N \leq 1000$ ). В последующих  $N$  строках идут фамилии кандидатов по одному в каждой строке. Каждая фамилия содержит от 1 до 10 букв английского алфавита, при этом первая буква прописная, а остальные – строчные.

### Выходные данные

В выходной файл OUTPUT.TXT выведите гистограмму голосования в форме прямоугольника, состоящего из символов «.» (ASCII 46) и «X» (ASCII 88). Число строк должно соответствовать максимальному количеству голосов, отданных за какого-либо кандидата. Число столбцов должно соответствовать количеству кандидатов, участвовавших в выборах. Гарантируется, что за каждого кандидата был отдан как минимум один голос. Высота  $i$ -го столбца (число символов «X») таблицы должна соответствовать числу отданных голосов за  $i$ -го кандидата. При этом кандидаты перечисляются в алфавитном порядке слева направо. Столбцы обозначаются символами «X» снизу вверх, пустые места – символами «.».

### Пример

| INPUT.TXT | OUTPUT.TXT |
|-----------|------------|
| 7         | .X..       |
| Ivanov    | .X.X       |
| Petrov    | XXXX       |
| Sidorov   |            |
| Petrov    |            |
| Zubov     |            |
| Petrov    |            |
| Zubov     |            |

## Задача С. Дендрохронология

*Ограничение по времени: 2 секунды*

*Ограничение по памяти: 32 Мб*

Дендрохронология – метод, в основе которого лежит закон природы, согласно которому каждый год толщина дерева увеличивается на одно кольцо. Толщина каждого кольца зависит от погоды в год образования кольца. У деревьев, растущих в одном и том же месте и климате толщина колец примерно одинакова. По числу колец можно определить возраст дерева, а по толщине колец – погодные условия каждого года, в который это дерево росло.

Используя «перекрестную датировку», можно построить дендрохронологическую шкалу путем увязывания воедино следующих друг за другом поколений деревьев, годы жизни которых перекрываются. Именно так возможно определить погодные условия от некоторого времени до настоящего момента. А на основании построенной шкалы можно датировать те или иные деревянные предметы, сохранившиеся с давних лет.

Однажды, в одном древнем городе, основанном в болотистой местности, археологи обнаружили  $N$  слоев мостовых, положенных один над другим и состоящих из отлично сохранившихся бревен. При этом первый слой был положен в момент основания города, а последний – в текущем году. Определив толщину колец бревен для каждого слоя мостовой, они задались вопросом: каков возраст этого города?

Ваша задача – помочь археологам вычислить возраст города по имеющимся данным.

### Входные данные

Первая строка входного файла INPUT.TXT содержит натуральное число  $N$  – количество слоев мостовых. Далее следует  $N$  строк, описывающих в хронологическом порядке дендрохронологическую шкалу для каждого бревна, взятого из отдельного слоя мостовой. Каждая строка состоит из цифр от 1 до 3, соответствующих толщине годовых колец в миллиметрах описываемого дерева от первого до последнего года его жизни. Суммарная длина всех строк не превышает  $10^6$ . Гарантируется, что дерево, из которого получено бревно следующего уровня, начало расти не раньше, чем дерево, из которого получено бревно предыдущего уровня. Аналогично гарантируется, что дерево, из которого получено бревно следующего уровня, было срублено не раньше, чем дерево, из которого получено бревно предыдущего уровня. Также известно, что годы жизни деревьев соседних уровней перекрываются. При неоднозначном определении общего периода жизни для деревьев смежных слоев следует выбрать максимально возможный диапазон.

### Выходные данные

В выходной файл OUTPUT.TXT выведите единственное целое число – возраст города.

### Пример

| INPUT.TXT  | OUTPUT.TXT | Пояснение  |
|--|------------|--|
| 4<br>123123123<br>3123111321<br>13212212<br>212331 | 13         | 123123123<br>.....3123111321<br>.....13212212<br>.....212331<br>(на основании этих данных можно определить дендрохронологическую шкалу для последних 22 лет) |

### Система оценивания

Решения, работающие только в тех случаях, когда общий период жизни деревьев в смежных слоях составляет не более 10 лет, будут оцениваться в 40 баллов.

## Задача D. Числа Фибоначчи

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 Мб

Последовательностью Фибоначчи называется бесконечная числовая последовательность целых чисел  $F_k$ , где  $F_0 = 0$ ,  $F_1 = 1$ ,  $F_k = F_{k-1} + F_{k-2}$  (для любого целого  $k$ ). Пример части данной последовательности:

|       |    |    |   |   |   |   |   |   |   |
|-------|----|----|---|---|---|---|---|---|---|
| k     | -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| $F_k$ | -1 | 1  | 0 | 1 | 1 | 2 | 3 | 5 | 8 |

Требуется найти остаток от деления  $n$ -го числа Фибоначчи на  $10^9$ .

Напомним, что остатком от деления целого числа  $a$  на целое число  $b$  называют такое неотрицательное целое число  $r$  ( $0 \leq r < |b|$ ), что выполняется следующее равенство:  $a = b \times q + r$ , где  $q$  – некоторое целое число.

### Входные данные

Входной файл INPUT.TXT содержит целое число  $n$  ( $-10^{18} \leq n \leq 10^{18}$ ).

### Выходные данные

В выходной файл OUTPUT.TXT выведите одно число – остаток от деления  $F_n$  на  $10^9$ .

### Примеры

| № | INPUT.TXT | OUTPUT.TXT |
|---|-----------|------------|
| 1 | 1         | 1          |
| 2 | 10        | 55         |
| 3 | -2        | 999999999  |

### Система оценивания

Решения, работающие для  $0 \leq n \leq 30$ , будут оцениваться в 20 баллов.

Решения, работающие для  $-10^6 \leq n \leq 10^6$ , будут оцениваться в 40 баллов.

Решения, работающие для  $-10^9 \leq n \leq 10^9$ , будут оцениваться в 80 баллов.

## Задача Е. Крестики-нолики

Ограничение по времени: 1 секунда

Ограничение по памяти: 16 Мб

Это интерактивная задача.

«Крестики-нолики» – популярная логическая игра на квадратном поле  $3 \times 3$  для двух игроков, которые по очереди ставят на свободные клетки знаки (первый ставит «крестики», второй – «нолики»). Выигрывает тот, кто первым поставит свои фигуры в ряд по вертикали, горизонтали или диагонали. Известно, что при выборе оптимальной стратегии обоими игроками игра сводится к ничьей.

Требуется написать программу, которая играет и не проигрывает программе жюри.

### Протокол взаимодействия

Сначала на вход программе подается целое число  $P$  ( $1 \leq P \leq 2$ ) – номер вашего хода: если  $P=1$ , то вы играете «крестиками», в противном случае – «ноликами». Далее идет последовательность обменов ходами вашей программы и программы жюри. Каждый ход описывается парой целых чисел  $X$  и  $Y$  ( $1 \leq X, Y \leq 3$ ) – номер столбца и строки, куда следует разместить очередной «крестик» или «нолик». При  $P=2$  после чтения значения  $P$  следует считывать первый ход программы жюри. Всего за игру первый игрок совершает 5 ходов, а второй – 4. Таким образом, первый игрок выполняет последний ход. В конце игры (когда вы сделаете последний свой ход) ваша программа должна немедленно завершиться.

Гарантируется, что программа жюри не проигрывает и придерживается оптимальной стратегии. Также гарантируется, что в каждом отдельном тесте при одинаковой последовательности ваших действий, программа жюри будет также повторять свою последовательность ходов.

### Примеры

| № | стандартный ввод                     | стандартный вывод               | пояснение         |
|---|--------------------------------------|---------------------------------|-------------------|
| 1 | 1<br>3 1<br>3 2<br>1 1<br>2 3        | 2 2<br>1 2<br>3 3<br>2 1<br>1 3 | OXO<br>XXO<br>XOX |
| 2 | 2<br>2 2<br>2 3<br>3 1<br>1 2<br>3 3 | 1 1<br>2 1<br>1 3<br>3 2        | OOX<br>XXO<br>OXX |

### Примечание

Для корректной работы программы после каждой операции вывода данных выводите перевод строки, а также очищайте буфер вывода. Очистка буфера вывода производится следующим образом:

- В языке Pascal: **flush(output)**
- В C/C++: **fflush(stdout)** или **cout.flush()**
- В Java: **System.out.flush()**
- В Python: **sys.stdout.flush()** из библиотеки **sys**
- В C# и Basic: **Console.Out.Flush()**

### Система оценивания

Решения, работающие только для  $P = 1$ , будут оцениваться в 25 баллов.