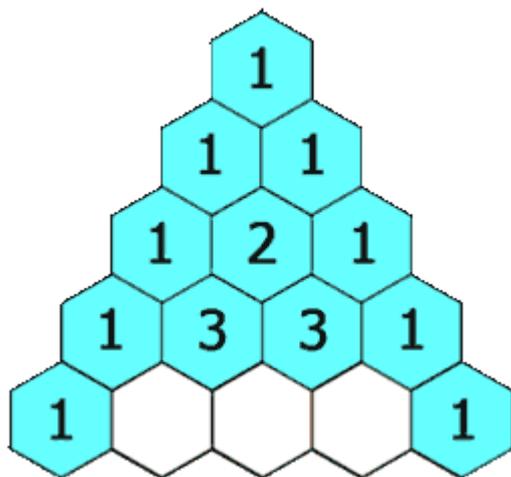


### Задача № 1

Учитывая целое число numRows, верните первые числа треугольника Паскаля. В треугольнике Паскаля каждое число является суммой двух чисел, расположенных непосредственно над ним, как показано на рисунке:



#### Пример № 1:

Ввод: numRows = 5

Вывод: [[1],[1,1],[1,2,1],[1,3,3,1],[1,4,6,4,1]]

#### Пример № 2:

Ввод: numRows = 1

Вывод: [[1]]

1 <= numRows <= 30

### Задача № 2

На вход методу подаётся строка message, которая имитирует сообщение игроков в различных онлайн играх. Игроки часто используют аббревиатуры, которые могут быть непонятны новичкам. Ваша задача, чтобы на выходе сообщение, полученное в результате обработки вашего метода, совпадало с исходным, но все аббревиатуры раскрываются в их полное описание.

Всего аббревиатур будет 6:

**2mor** – tomorrow

**2ez** – too easy

**aap** – always a pleasure

**asap** – as soon as possible

**dur** – do you remember

**imho** – in my humble opinion

Пример работы метода:

I think it was 2ez -> I think it was too easy.

2ezasap -> Too easy as soon as possible

It was a busy day. -> It was a busy day.

DUR MAN ABOUT 2MOR? -> Do you remember MAN ABOUT tomorrow?

Сообщения могут подаваться в разном регистре, но результат раскрытия аббревиатур производится в соответствии с расположением. Если сообщение начинается с аббревиатуры, то первый символ обязан быть в верхнем регистре, остальные согласно сообщению.

### Задача № 3

Определите, подходит ли доска для sudoku размером 9 x 9. Только заполненные ячейки должны быть проверены в соответствии со следующими правилами: Каждая строка должна содержать цифры от 1 до 9 без повторений. Каждый столбец должен содержать цифры от 1 до 9 без повторений. Каждая из девяти ячеек сетки размером 3 x 3 должна содержать цифры от 1 до 9 без повторений. Примечание: Доска для sudoku (частично заполненная) может быть действительной, но не обязательно разрешимой. Только заполненные ячейки должны быть проверены в соответствии с указанными правилами.

#### Example 1:

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Ввод: board =

```
[["5","3",,"","7",,"","",""],  
["6",,"","1","9","5",,"",""],  
[,"9","8",,"","","","6",,""],  
["8",,"","","6",,"","","3"],  
["4",,"","8",,"3",,"","1"],  
["7",,"","","2",,"","","6"],  
[,"6",,"","","2","8",,""],  
[,"","4","1","9",,"","5"],  
[,"","","8",,"","7","9"]]
```

Вывод: true

#### Пример 2:

Ввод: board =

```
[["8","3",,"","7",,"","",""],  
["6",,"","1","9","5",,"",""],  
[,"9","8",,"","","","6",,""],  
["8",,"","","6",,"","","3"],  
["4",,"","8",,"3",,"","1"],  
["7",,"","","2",,"","","6"],  
[,"6",,"","","2","8",,""],  
[,"","4","1","9",,"","5"],  
[,"","","8",,"","7","9"]]
```

Вывод: false

Пояснение: То же, что и в примере 1, за исключением того, что цифра 5 в верхнем левом углу изменена на 8. Поскольку в верхнем левом поле размером 3x3 есть две цифры 8, это недопустимо.

#### Задача № 4

На вход программе подаётся отсортированный массив целых чисел. Ваша задача на выходе вернуть рассортированный массив таким образом, чтобы ближайшие по значению числа(с разницей в 1) не располагались по соседству(слева и справа друг от друга). Если это невозможно или массив состоит из одного числа или пустой, то метод возвращает пустой массив.

Пример:

```
[1,2,3,4,5] -> [5, 2, 4, 3, 1]
[5,4,3,2,1] -> [2, 4, 3, 1, 5]
[1,1,1,1] -> []
[2,3,4,5] -> []
[] -> []
```

#### Задача № 5

У Василия есть число  $a$ , которое он хочет превратить в число  $b$ . Для этого он может производить два типа операций:

- умножить имеющееся у него число на 2 (то есть заменить число  $x$  числом  $2 \cdot x$ );
- приписать к имеющемуся у него числу цифру 1 справа (то есть заменить число  $x$  числом  $10 \cdot x + 1$ ).

Вам надо помочь Василию получить из числа  $a$  число  $b$  с помощью описанных операций, либо сообщить, что это невозможно. Обратите внимание, что в этой задаче не требуется минимизировать количество операций. Достаточно найти любой из способов получить из числа  $a$  число  $b$ .

#### Задача № 6

В качестве сувениров на Олимпиаде по программированию было решено вручить футболки. Всего в типографии были напечатаны футболки шести размеров: S, M, L, XL, XXL, XXXL (размеры перечислены в порядке возрастания). Для каждого размера от S до XXXL вам известно количество футболок такого размера.

Во время регистрации организаторы попросили каждого из  $n$  участников указать размер футболки. Если участник колебался между двумя размерами, то он мог указать два соседних — это означает, что ему подойдет футболка любого из двух размеров.

Напишите программу, которая определит, возможно ли из напечатанных в типографии футболок сделать подарок каждому участнику соревнования. Конечно, каждому участнику должна достаться футболка его размера:

- требуемого размера, если указан один размер;
- любого из двух размеров, если указаны два соседних размера.

В случае положительного ответа программа должна найти любой из вариантов раздачи футболок.

### Задача № 7

Образовательная организация в очередной раз проводит Олимпиаду по программированию.

Заботясь о персональных данных участников, организация придумала хитрый алгоритм шифрования:

Подсчитывается количество различных символов в ФИО (регистр важен, А и а — разные символы). Берётся сумма цифр в дне и месяце рождения, умноженная на 64.

Для первой (по позиции в слове) буквы фамилии определяется её номер в алфавите (в 1-индексации), умноженный на 256 (регистр буквы не важен). Полученные числа суммируются.

Результат переводится в 16-чную систему счисления (**в верхнем регистре**). У результата сохраняются только 3 младших разряда (если значимых разрядов меньше, то шифр дополняется до 3-х разрядов ведущими нулями).

Ваша задача — помочь вычислить для каждого кандидата его шифр.

Рассмотрим тестовый пример.

Первый кандидат — Volozh, Arcady, Yurievich, 11, 2, 1964:

Различные символы в ФИО: V, o, l, z, h, A, r, c, a, d, y, Y, u, i, e, v - всего их **16**.

Сумма цифр в дне и месяце рождения равна

$$1+1+2=4.$$

Номер в алфавите первой буквы фамилии V равен **22**.

Итоговое значение шифра равно

$$16+4\cdot 64+22\cdot 256=5904.$$

В 16-ричной системе счисления это число представимо как **1710**.

Нас интересуют только 3 последние разряда, поэтому остаётся **710**.

Второй кандидат — Segalovich, Ilya, Valentinovich, 13, 9, 1964:

Различные символы в ФИО: S, e, g, a, l, o, v, i, c, h, I, y, V, n, t - всего их **15**.

Сумма цифр в дне и месяце рождения равна

$$1+3+9=13.$$

Номер в алфавите первой буквы фамилии S равен **19**.

Итоговое значение шифра равно

$$15+13\cdot 64+19\cdot 256=5711.$$

В 16-ричной системе счисления это число представимо как **164F**.

Нас интересуют только 3 последние разряда, поэтому остаётся **64F**.