**Задачи:**

1. [**https://leetcode.com/problems/powx-n/**](https://leetcode.com/problems/powx-n/)
2. **Дана последовательность из N целых чисел. Найти сумму чисел, оканчивающихся на 5, перед которыми идет четное число.**
3. **Дан массив целых чисел. Найти количество пар соседних элементов, где первый элемент вдвое больше второго.**
4. **Задача на ООП:**

1) Проектирование и создание класса, описывающего [вектор](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%BA%D1%82%D0%BE%D1%80_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0))

Задача:  
  
Создайте класс, который описывает вектор (в трёхмерном пространстве).  
  
У него должны быть:

* конструктор с параметрами в виде списка координат *x, y, z*
* метод, вычисляющий длину вектора. Корень можно посчитать с помощью Math.sqrt():



* метод, вычисляющий скалярное произведение:



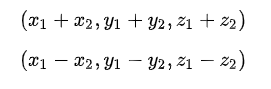
* метод, вычисляющий векторное произведение с другим вектором:



* метод, вычисляющий угол между векторами (или косинус угла): косинус угла между векторами равен скалярному произведению векторов, деленному на произведение модулей (длин) векторов:



* методы для суммы и разности:



1. **\* Отвалидировать доску судоку**

/\*Требования

Объявлена доска 9 x 9 необходимо отвалидировать в соответствии с правилами

1. Каждая строка должна содержать цифру 1-9 без повторения

2. Каждая колонка должна содержать цифру 1-9 без повторения

3. Каждый под блок из 9 элементов 3 x 3 должна содержать цифру 1-9 без повторения

Ограничения

• board.length == 9

• board[i].length == 9

• board[i][j] значение число или '.'

\*/

public class Main {

public static void main(String[] args) {

char[][] 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'}};

System.out.println("Validete:" + isValidSudoku(board));

}