**Подготовка к тестированию**

*Вектор*

1. Программой генерируются числа в диапазоне [0;10000]. Необходимо написать программу, которая поместит все четные числа в начало вектора, а нечетные в конец вектора, выведет все элементы вектора в строку, а также найдет минимальное четное число, в составе которого нет цифры 0, выведет это число во вторую строку, если таких чисел несколько, выбрать любое из них, если такого числа нет, то в качестве ответа вывести “No”.

**Пример**

Input: 12 17 20 3 10 9 128 77 64 520 7 -1

Output**:** 520 64 128 10 20 12 17 3 9 77 7

12

1. С клавиатуры вводятся числа в диапазоне от 10 до100 (условие должно проверяться программой). Количество чисел не известно. Ввод заканчивается -1 Необходимо написать программу, которая запишет в новый вектор в порядке возрастания все числа, у которых вторая цифра (справа налево) нечетная. Если таких чисел нет, то программа выведет файл информацию об ошибке.

*Подпрограммы*

1. С помощью подпрограммы найти периметр треугольника. В основной программе вводятся координаты вершин треугольника и находится его периметр. Длины отдельных сторон находятся с помощью подпрограммы.

**Точка 1: (x1; y1) | Точка 2: (x2; y2)**

**A = √(X²+Y²) = √((X2-X1)²+(Y2-Y1)²)**.

1. Составить функцию, которая для одномерного массива *K* найдет сумму элементов, в составе которых есть цифра “5”.
2. Написать подпрограмму *print* для вывода матрицы на экран. Будет учитываться способ передачи вектора в функцию.
3. Двоичное число, введенное пользователем программы, преобразовать в десятичное число. Результат вывести на экран.
4. Написать рекурсивную процедуру, переводящую числа из одной системы счисления в другую.

*Строки*

1. Задан действительный (IPv4) IP-адрес, верните измененную версию этого IP-адреса.

**Пример**

Input: "1.1.1.1"

Output**:** "1[.]1[.]1[.]1"

1. ***Предложение*** – это список слов, разделенных одним пробелом, без ведущих и последующих пробелов. Вам дан массив строк sentences, где каждая sentences[i] представляет собой одно предложение. Верните ***максимальное количество слов***, встречающихся в одном предложении.

**Пример**

Input: sentences = ["alice and bob love leetcode", "i think so too", "this is great thanks very much"]

Output**:** 6

1. Дана строка *s*, верните строку после замены каждой прописной буквы на такую же строчную.
2. Вам дана строка *s* и целочисленный массив *spaces*, который описывает индексы в исходной строке, куда будут добавлены пробелы. Каждый пробел должен быть вставлен перед символом с заданным индексом. Например, учитывая s = "EnjoyYourCoffee" и spaces = [5, 9], мы помещаем пробелы перед 'Y' и 'C', которые находятся в индексах 5 и 9 соответственно. Таким образом, мы получаем "Enjoy Your Coffee". Верните модифицированную строку после добавления пробелов.

*Указатели*

1. В матрице размером *N* необходимо заменить элементы, которые по значению меньше произведения номера строки и столбца в которых находится элемент, на ноль. Вывести на экран количество замен, новую матрицу.