Программы, решающие задачи, разрабатываются на Java (Java 8, 13 или 17 – не

принципиально).

Достаточно решить задачи 1, 2 (обе) и одну из задач 3 или 4 на выбор (или обе, если хочется

предъявить высокий уровень).

1. Заполнить массив случайными целыми числами. Вывести массив на экран. Переупорядочить

в этом массиве элементы следующим образом: сначала по не убыванию нечетные числа,

потом нули, потом прочие числа по не возрастанию. Вывести массив на экран.

2. Найти в массиве наиболее часто встречающееся число (числа, если таких несколько),

вывести на экран исходные данные и результаты.

3. Играют 2 игрока. Каждый из них перед игрой тайно от другого игрока выбирает по одной

последовательности из 3 чисел от 1 до 6 (числа могут совпадать). Далее они по очереди

бросают кубик определенное число раз (100, 1000, ...). Каждый из игроков получает в игре

столько очков, сколько раз выпала последовательность из тех чисел, которые он выбрал. При

этом последовательности чисел одного игрока (которые приносят ему очки) не должны

пересекаться, а последовательности чисел разных игроков могут пересекаться.

Поясняющие примеры.

Допустим у игрока 1 последовательность 4, 2, 4, у игрока 2 последовательность 4, 4, 4. Кубик

бросается 10 раз. Выпало 1, 4, 2, 4, 4, 4, 4, 4, 2, 4. Игрок 1 набрал 2 очка, игрок 2 набрал 1 очко.

Допустим у игрока 1 последовательность 4, 2, 4, у игрока 2 последовательность 4, 4, 4. Кубик

бросается 10 раз. Выпало 1, 4, 2, 4, 4, 4, 4, 4, 4, 4. Игрок 1 набрал 1 очко, игрок 2 набрал 2 очка.

Допустим у игрока 1 последовательность 4, 2, 4, у игрока 2 последовательность 2, 4, 2. Кубик

бросается 12 раз. Выпало 4, 2, 4, 2, 4, 2, 4, 2, 4, 2, 4, 1. Игрок 1 набрал 3 очка, игрок 2 набрал 2

очка.

Допустим у игрока 1 последовательность 1, 2, 3, у игрока 2 последовательность 2, 3, 1. Кубик

бросается 12 раз. Выпало 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3, 1. Игрок 1 набрал 3 очка, игрок 2 набрал 4

очка.

Допустим у игрока 1 последовательность 1, 2, 3, у игрока 2 последовательность 4, 5, 5. Кубик

бросается 12 раз. Выпало 1, 2, 4, 3, 5, 5, 2, 3, 4, 5, 5, 5. Игрок 1 набрал 0 очков, игрок 2 набрал 1

очко.

Необходимо для двух фиксированных наборов 3 чисел для каждого из игроков и

фиксированного числа бросков кубика вычислить для обоих игроков вероятность набора

игроком большего, чем у соперника, числа очков, а также вероятность ничьей. Достаточно

приближенного вычисления, допустимо применить метод Монте-Карло.

4. Дан набор целых чисел (возможны повторяющиеся числа) и число K. Написать программу, которая разделяет данный набор на K частей, удовлетворяющих следующему правилу: сумма чисел в одной из частей равна L, во второй L + 1 и т.д., сумма чисел в последней части равна L + K - 1, где L – некоторое число.

Вывести на экран исходные данные и ответ (части и суммы чисел в этих частях). Поясняющие примеры. [10, 11, 7, 7, 12], K = 2. Программа в качестве ответа выводит: [11, 12], 23, [10, 7, 7], 24. [5, 2, 6, 4, 3, 6], K = 4. Программа в качестве ответа выводит: [5], 5, [6], 6, [4, 3], 7, [2, 6] 8. [7, 8, 12, 1], K = 3. Программа в качестве ответа выводит: невозможно.