

Для выполнения заданий перейдите по ссылке из письма, скачайте файл выбранного языка программирования и выполняйте задание в нем БЕЗ ИЗМЕНЕНИЯ архитектуры, названий классов, функций и методов.

Если Вы не нашли файл для своего любимого языка, то выполняйте задания строго придерживаясь входных / выходных данных в созданном Вами проекте с нуля.

Решение всех задач присылать **строго одним файлом** (.cs, .js, .py и др.). Допускается прямая ссылка на файл, который лежит на файлообменнике или в GIT.

### Задание 1

#### Сумма факториалов всех простых чисел, не превышающих X

Входные данные:

- 1) Число X (Integer / Целочисленный)

Выходные данные:

- 1) Сумма (Integer / Целочисленный)

Требуется написать алгоритм, вычисляющий сумму факториалов всех простых чисел, не превышающих число X.

Справка: простым числом называют целое положительное число, которое имеет только два различных натуральных делителя – единица и само число. Примеры простых чисел: 2, 3, 5, 7, 11, ...

Например, введено число 5. Программа должна вывести 128 ( $2! + 3! + 5!$ ).

### Задание 2

#### ПО для вендингового аппарата

Входные данные:

- 1) Сумма заказа (Integer / Целочисленный);
- 2) Внесённая сумма клиентом (Integer / Целочисленный).

Выходные данные:

- 1) Сдача формата «номинал» : «количество» (Dictionary / Словарь)

Клиент выбирает товар в вендинговом аппарате (автоматизированном аппарате продажи товаров) и вносит сумму для оплаты заказа. Считаем, что клиент вносит всегда сумму большую или равную сумме заказа.

Требуется написать алгоритм, вычисляющий количество денег каждого номинала, необходимых выдать клиенту в качестве сдачи. Для выдачи сдачи можно использовать только купюры и монеты следующего номинала: 1, 2, 5, 10, 50, 100, 200, 500, 1000, 2000, 5000 рублей. Большие купюры имеют выше приоритет к выдаче, по сравнению с меньшими.

Например, товар стоит 41 руб. Клиент внёс 88 руб. Сдача клиенту 47 руб, программа должна вывести словарь: { 10: 4; 5: 1; 2: 1 } (10 руб 4 шт., 5 руб 1 шт., 2 руб 1 шт.)

### Задание 3

#### Заказы на линзы для Инопланетян

Входные данные:

- 1) Массив чисел, каждый элемент которого указывает на рекомендованную оптическую силу линзы для каждого i-го Инопланетянина.

Выходные данные:

- 1) Количество пар линз (Integer / Целочисленный)

На выдуманной нами планете «Железяка» живут **одноглазые** Инопланетяне. Некоторые Инопланетяне жалуются на зрение, поэтому им рекомендовано носить контактные линзы. При этом, если Инопланетянину рекомендовано ношение линз в X диоптрий, его устроят линзы в X-1, X, X+1 диоптрии. Инопланетяне достают линзы посещая

магазины на планете «Земля», где они продаются только парами по две штуки. Самый инициативный Инопланетянин решил оптимизировать процесс закупки линз таким образом, чтобы покупать как можно меньше пар и при этом удовлетворять все поступившие к нему заказы.

Требуется написать алгоритм, вычисляющий минимальное количество пар линз, необходимых для удовлетворения всех поступивших заказов.

Например, для 5-ти Циклопов с диоптриями [1, -1, 2, 3, -3], алгоритм должен посчитать 3 пары линз (0 диоптрий удовлетворяют 1-го и 2-го из массива Инопланетян, 2 или 3 диоптрия удовлетворяют 3-го и 4-го, -2 или -3 или -4 диоптрии удовлетворяют 5-го, но 1 линза останется лишняя в паре — это нормально)

**Желаем удачи!**