**Homework 1**

1. Напишите программу, которая считает общую цену. Вводится M рублей и N копеек цена, а также количество L товара Посчитайте общую цену в рублях и копейках за L товаров.

Пример:

Input: Цена одной вещи 3 рубля 20 копеек, посчитать 3 предмета.

Output: Общая цена 9 рублей 60 копеек

2. Зарегистрируйтесь на одном (или нескольких) из сайтов:

https://py.checkio.org/ , https://www.codewars.com,   https://www.hackerrank.com/, https://acmp.ru И решите 1-5 задач уровня Elementary или любых других. Результаты поместите в Пул реквест. (Можно больше :))

3. Найти самое длинное слово в введенном предложении. Учтите что в предложении есть знаки препинания.

Подсказки:

* my\_string.split([chars]) возвращает список строк.
* len(list) - количество элементов в списке

4. Вводится строка. Требуется удалить из нее повторяющиеся символы и все пробелы. Например, если было введено "abc cde def", то должно быть выведено "abcdef".

5. Посчитать количество строчных (маленьких) и прописных (больших) букв в введенной строке. Учитывать только английские буквы.

Дополнительные задания:

- Выведите n-ое число Фибоначчи, используя только временные переменные, циклические операторы и условные операторы. n - вводится

- Определите, является ли число палиндромом (читается слева направо и справа налево одинаково).  Число положительное целое, произвольной длины. Задача требует работать только с числами (без конвертации числа в строку или что-нибудь еще)

**Homework 2**

1. **FizzBuzz**

Напишите программу, которая печатает цифры от 1 до 100, но вместо чисел, кратных 3 пишет Fizz, вместо чисел кратный 5 пишет Buzz, а вместо чисел одновременно кратных и 3 и 5 - FizzBuzz

1. **List practice**
2. Используйте генератор списков чтобы получить следующий: ['ab', 'ac', 'ad', 'bb', 'bc', 'bd'].
3. Используйте на предыдущий список slice чтобы получить следующий: ['ab', 'ad', 'bc'].
4. Используйте генератор списков чтобы получить следующий ['1a', '2a', '3a', '4a'].
5. Одной строкой удалите элемент  '2a' из прошлого списка и напечатайте его.
6. Скопируйте список и добавьте в него элемент '2a' так чтобы в исходном списке этого элемента не было.
7. **Tuple practice**
8. Создайте список ['a', 'b', 'c'] и сделайте из него кортеж.
9. Создайте кортеж ('a', 'b', 'c'), И сделайте из него список
10. Сделайте следующие присвоения одной строкой a = 'a', b=2, c=’python’.
11. Создайте кортеж из одного элемента, чтобы при итерировании по этому элементы последовательно выводились значения 1, 2, 3. Убедитесь что len() исходного кортежа возвращает 1.
12. **Пары элементов**
13. Дан список чисел. Посчитайте, сколько в нем пар элементов, равных друг другу. Считается, что любые два элемента, равные друг другу образуют одну пару, которую необходимо посчитать.
14. Входные данные - строка из чисел, разделенная пробелами.
15. Выходные данные - количество пар.
16. Важно: 1 1 1 - это 3 пары, 1 1 1 1 - это 6 пар
17. **Уникальные элементы в списке**

Дан список. Выведите те его элементы, которые встречаются в списке только один раз. Элементы нужно выводить в том порядке, в котором они встречаются в списке.

1. **Упорядоченный список.**

Дан список целых чисел. Требуется переместить все ненулевые элементы в левую часть списка, не меняя их порядок, а все нули - в правую часть. Порядок ненулевых элементов изменять нельзя, дополнительный список использовать нельзя, задачу нужно выполнить за один проход по списку. Распечатайте полученный список.

**Homework 3**

1. Dict comprehensions

Создайте словарь с помощью генератора словарей, так чтобы его ключами были числа от 1 до 20, а значениями кубы этих чисел.

1. Города

Дан список стран и городов каждой страны. Затем даны названия городов. Для каждого города укажите, в какой стране он находится.

Входные данные

Программа получает на вход количество стран *N*. Далее идет *N* строк, каждая строка начинается с названия страны, затем идут названия городов этой страны. В следующей строке записано число *M*, далее идут *M* запросов — названия каких-то *M* городов, перечисленных выше.

Выходные данные

Для каждого из запроса выведите название страны, в котором находится данный город.

Примеры

Входные данные

2

Russia Moscow Petersburg Novgorod Kaluga

Ukraine Kiev Donetsk Odessa

3

Odessa

Moscow

Novgorod

Выходные данные

Ukraine

Russia

Russia

1. Даны два списка чисел. Посчитайте, сколько различных чисел содержится одновременно как в первом списке, так и во втором.
2. Даны два списка чисел. Посчитайте, сколько различных чисел входит только в один из этих списков
3. Языки

Каждый из *N* школьников некоторой школы знает *Mi* языков. Определите, какие языки знают все школьники и языки, которые знает хотя бы один из школьников.

**Входные данные**

Первая строка входных данных содержит количество школьников *N*. Далее идет *N* чисел *Mi*, после каждого из чисел идет *Mi* строк, содержащих названия языков, которые знает *i*-й школьник. Пример:

3          - N количество школьников

2          - M1 количество языков первого школьника

Russian    - языки первого школьника

English

3          - M2 количество языков второго школьника

Russian

Belarusian

English

3

Russian

Italian

French

**Выходные данные**

В первой строке выведите количество языков, которые знают все школьники. Начиная со второй строки - список таких языков. Затем - количество языков, которые знает хотя бы один школьник, на следующих строках - список таких языков.

1. Слова

Во входной строке записан текст. Словом считается последовательность непробельных символов идущих подряд, слова разделены одним или большим числом пробелов или символами конца строки. Определите, сколько различных слов содержится в этом тексте.

Количество максимальных баллов 9

На 9+:

1. Оглянемся назад

Даны два натуральных числа. Вычислите их наибольший общий делитель при помощи алгоритма Евклида (мы не знаем функции и рекурсию).

**Homework 4**

1. Оформите решение задач из прошлых домашних работ в функции. Напишите функцию runner. (все станет проще когда мы изучим модули, getattr и setattr)
   1. runner() – все фукнции вызываются по очереди
   2. runner(‘func\_name’) – вызывается только функцию func\_name.
   3. runner(‘func’, ‘func1’...) - вызывает все переданные функции
2. Создайте декоратор, который хранит результаты вызовы функции (за все время вызовов, не только текущий запуск программы)
3. Реализовать функцию get\_ranges которая получает на вход непустой список неповторяющихся целых чисел, отсортированных по возрастанию, которая этот список “сворачивает”

 get\_ranges([0, 1, 2, 3, 4, 7, 8, 10]) *// "0-4,7-8,10"*

 get\_ranges([4,7,10]) *// "4,7,10"*

 get\_ranges([2, 3, 8, 9]) *// "2-3,8-9"*

1. В файле хранятся данные с сайта IMDB. Скопированные данные хранятся в файле ./data5/ ratings.list.
   1. Откройте и прочитайте файл(если его нет необходимо вывести ошибку).
   2. Найдите ТОП250 фильмов и извлеките заголовки.
   3. Программа создает 3 файла  top250\_movies.txt – названия файлов, ratings.txt – гистограмма рейтингов, years.txt – гистограмма годов.
2. Написать программу которая находит ближайшую степень двойки к введенному числу. 10(8), 20(16), 1(1)
3. Вводится число найти его максимальный делитель, являющийся степенью двойки. 10(2) 16(16), 12(4).

**Homework 5**

1. Создайте  модель из жизни. Это может быть бронирование комнаты в отеле, покупка билета в транспортной компании, или простая РПГ. Создайте несколько объектов классов, которые описывают ситуацию Объекты должны содержать как атрибуты так и методы класса для симуляции различных действий. Программа должна инстанцировать объекты и эмулировать какую-либо ситуацию - вызывать методы, взаимодействие объектов и т.д.
2. Создайте декоратор, который вызывает задекорированную функцию пока она не выполнится без исключений (но не более n раз - параметр декоратора). Если превышено количество попыток, должно быть возбуждено исключение типа TooManyErrors