

Чтение данных во всех задачах осуществляется из файла

1. Напишите функцию для нахождения НОД произвольного количества чисел.
2. Создайте функцию, которая выполняет умножение двух матриц и возвращает результат в виде новой матрицы.
3. Создайте функцию, которая преобразует матрицу в список её элементов, проходя по матрице зигзагообразно (например, сверху вниз, затем снизу вверх и так далее)
4. Реализуйте функцию, которая транспонирует матрицу (двумерный список). Функция должна возвращать новую матрицу, не меняя оригинал.
5. Создайте рекурсивную функцию, которая принимает список списков и возвращает один "плоский" список, содержащий все элементы вложенных списков.
6. Создайте функцию, которая "сглаживает" матрицу, заменив каждый элемент на среднее арифметическое его соседей и себя.
7. Напишите функцию, которая принимает матрицу и значение, а затем ищет это значение в матрице и возвращает список пар индексов, где это значение встречается (или пустой список, если таких нет).
8. Создайте функцию, которая выделяет из списка чисел только четные и возвращает их в виде нового списка.
9. Создайте функцию, которая принимает список и функцию, которая применяется к каждому его элементу, результатом должен стать новый список с измененными элементами.
10. На вход подаётся список чисел. Реализовать функцию, которая преобразует список в максимально возможное число. Например, список [52, 8, 100] должен быть преобразован в число 852100. Результат вывести на экран.
11. Создайте функцию, которая проверяет, является ли двумерный массив магическим квадратом (суммы чисел в каждом ряду, каждом столбце и каждой диагонали одинаковы).
12. Создайте функцию, которая принимает произвольное количество списков и комбинирует их, чередуя элементы каждого списка (например, для списков 1, 2 и 'a', 'b' должно получиться 1, 'a', 2, 'b'). В случае разной длины списков, пропущенные значения должны быть заменены на «None».
13. Определите функцию `is_prime`, которая будет проверять число на простоту. Затем, создайте генератор `generate_primes`, который будет использовать глобальную переменную, хранящую последнее найденное

простое число, и при каждом вызове генерировать следующее простое число.

14. Напишите функцию-генератор, который создает все возможные перестановки переданного ей списка, используя рекурсивный алгоритм.

15. На вход подаётся список чисел и число N. Между числами списка можно ставить знаки «+», «-» или не ставить ничего. Написать функцию, которая выводит все выражения, составленные из чисел списка и знаков «+», «-», которые равны числу N.

16. Напишите функцию, которая принимает два множества и возвращает множество, представляющее их декартово произведение. Используйте генератор для создания пар элементов.

17. Напишите функцию, которая принимает список словарей, меняет ключи и значения местами, группирует общие для одинаковых ключей значения в списки и возвращает получившийся словарь.

18. Реализуйте функцию, которая принимает матрицу, представляющую карту, где 0 – вода, а 1 – земля. Функция должна подсчитать количество отдельных островов.