Functions

Write a function is\_prime(n) that takes an integer n and returns True if it is a prime number, otherwise False.

Create a function factorial(n) that calculates and returns the factorial of a number n using recursion.

Write a function count\_vowels(s) that takes a string s and returns the number of vowels (a, e, i, o, u) in the string.

Create a function reverse\_string(s) that returns the reverse of a given string s without using any built-in reverse functions.

Lists

Write a program to find the largest number in a list of integers.

Create a function remove\_duplicates(lst) that takes a list and returns a new list with duplicate elements removed.

Write a program to merge two lists and sort the merged list in ascending order.

Create a function rotate\_list(lst, k) that rotates a list lst by k steps to the right. For example, rotate\_list([1, 2, 3, 4, 5], 2) should return [4, 5, 1, 2, 3].

Sets

Write a program to find the common elements between two sets.

Create a function is\_subset(set1, set2) that checks if set1 is a subset of set2 and returns True or False.

Write a program to remove all elements from a set that are also present in another set.

Create a function unique\_elements(lst) that takes a list and returns a set of unique elements.

Tuples

Write a program to find the index of an element in a tuple.

Create a function swap\_tuple(t) that takes a tuple t and returns a new tuple with the first and last elements swapped.

Write a program to concatenate two tuples and return the result.

Create a function count\_occurrences(t, element) that counts how many times element appears in tuple t.

Dictionaries

Write a program to merge two dictionaries into one. If there are duplicate keys, the values from the second dictionary should overwrite the first.

Create a function invert\_dict(d) that takes a dictionary d and returns a new dictionary with keys and values swapped.

Write a program to count the frequency of each character in a string using a dictionary.

Create a function dict\_to\_list(d) that converts a dictionary into a list of tuples, where each tuple is a key-value pair.

File Handling

Write a program to read a text file and count the number of lines in it.

Create a function write\_to\_file(filename, content) that writes a string content to a file named filename.

Write a program to read a file and print only the lines that contain a specific word.

Create a function copy\_file(source, destination) that copies the contents of one file (source) to another file (destination).

Write a program to read a CSV file and store its contents in a list of dictionaries, where each dictionary represents a row.

Mixed Exercises

Write a function list\_to\_dict(lst) that converts a list of tuples into a dictionary, where the first element of each tuple is the key and the second element is the value.

Create a program that reads a text file, removes all punctuation, and writes the cleaned text to a new file.

Write a function find\_common\_words(file1, file2) that finds and returns the common words between two text files.

Create a program that takes a list of dictionaries (e.g., [{'name': 'Alice', 'age': 25}, {'name': 'Bob', 'age': 30}]) and writes it to a CSV file.

Write a function sort\_dict\_by\_value(d) that takes a dictionary and returns a list of keys sorted by their corresponding values.