**Exercises**

Problem 1- Write a function **symmetric\_difference\_count** that takes two sets and returns the count of elements that are unique to each set (symmetric difference) without using the built-in symmetric\_difference method.

Problem 2. Create a function **is\_disjoint\_sets** that takes two sets and returns True if they are disjoint (have no elements in common) and False otherwise. Don't use any built-in function.

Problem 3. You are given a list of sets, each representing the preferences of a group of people for a particular item. Your task is to find the item that maximizes the overall satisfaction across all groups. The satisfaction of each group for an item is given by the sum of the sizes of the intersection of the group's preference set with the selected item's set.

Problem 4. You are given a dictionary of sets containing vertices of a graph with an adjacency list. Your job is to find the cycles in the graph (if present).

If the graph is,

undirected\_graph = {

'A': {'B', 'C'},

'B': {'A', 'D'},

'C': {'A', 'D'},

'D': {'B', 'C'}

}

The graph has a cycle, write a program to implement the logic.

Problem 5. Write a function **calculate\_average** that takes a list of numerical values as input and calculates the average. However, the function should handle the following exceptions:

If the input list is empty, raise a ValueError with the message "List is empty. Cannot calculate the average."

If any element in the list is not a numerical value, raise a TypeError with the message "Invalid element type. All elements must be numerical."

Problem 6. You are given a text file named "grades.txt" with the following format:

Alice,95

Bob,87

Charlie,92

David,78

Eva,96

Your program should perform the following

1. Calculate and display the average grade of all students.
2. Identify and display the name of the student with the highest grade.