Lab exercises

Exercise 1

You are given a list of student names along with their scores in a particular subject. Write a C++ program to store this information in a map where the key is the student name and the value is the score. Then, display the student names along with their scores in ascending order of scores.

```
Input:

The first line contains an integer N, the number of students.
The next N lines contain the student name (a string without spaces) and the score (an integer
    separated by a space).
Output:

Display the student names along with their scores in ascending order of scores.
```

Exercise 2

You are given a list of words. Write a C++ program to count the frequency of each word and store it in a map where the key is the word and the value is the frequency. Then, display the words along with their frequencies in ascending order of frequencies.

```
Input:
6
apple
banana
apple
orange
banana
apple

Output:
orange 1
banana 2
apple 3
```

Exercise 3

You are given a list of integers. Write a C++ program to find the frequency of each integer and store it in an std::unordered_map, where the key is the integer and the value is the frequency. Then, display the integers along with their frequencies in ascending order of integers.

Input:

The first line contains an integer N, the number of integers. The next N lines contain an integer each.
Output:

Display the integers along with their frequencies in ascending order of integers.

```
Sample input:
8
5
```

```
2
8
2
6
8
5
3

Output:
2 2
3 1
5 2
6 1
8 3
```

Exercise 4 ....

You are given a list of student names along with their scores in multiple subjects. Each student can have scores in different subjects. Write a C++ program to store this information in both an std::map and an std::unordered_map, where the key is the student name and the value is a map of subject names to scores. Then, display the student names along with their total scores in ascending order of total scores using the std::map, and in any order using the std::unordered_map.

Input:

The first line contains an integer N, the number of students. For each student, the input consists of: A line containing the student name (a string without spaces). A line containing an integer M, the number of subjects the student has scores for. M lines, each containing a subject name (a string without spaces) and the score (an integer separated by a space). Output:

Display the student names along with their total scores in ascending order of total scores using the std::map, and in any order using the std::unordered_map.

```
input:
3
Ali
2
Math 85
Science 90
Baber
3
Math 90
Science 85
History 80
khan
1
Science 95

output
Baber 255
Ali 175
khan 95
```

```
output
Ali 175
Baber 255
khan 95
```

Exercise 5 . . . .

You are given a list of products along with their prices in a store. Each product has a unique ID, and the prices can change frequently. Write a C++ program to store this information in both an std::map and an std::unordered_map, where the key is the product ID and the value is the price. Then, simulate a price change by updating the price of a specific product. Finally, display the prices of all products in ascending order of product IDs using the std::map, and in any order using the std::unordered_map.

Input:

The first line contains an integer N, the number of products. For each product, the input consists of: A line containing the product ID (an integer). A line containing the price of the product (a floating-point number). After the initial prices are provided, the program should simulate a price change:

A line containing an integer M, the product ID for which the price needs to be updated. A line containing the new price for the product.

Output:

Display the prices of all products in ascending order of product IDs using the std::map, and in any order using the std::unordered_map.

```
sample input:
4
101
10.5
102
15.25
103
20.0
104
8.75
102
13.5

Sample output:
101 10.5
102 13.5
103 20.0
104 8.75

Sample output:
101 10.5
103 20.0
104 8.75
102 13.5
```

****