OOP Midterm – Design a set for string

Description

In this question, a class named StringSet will be defined to store a set of strings. You have the option to implement the functions of this class using either an array or a vector.

Given a main.cpp file and a StringSet.h header file, your task is to implement the member functions of the StringSet class in the StringSet.cpp file.

Below are the member functions to be implemented in the StringSet class.

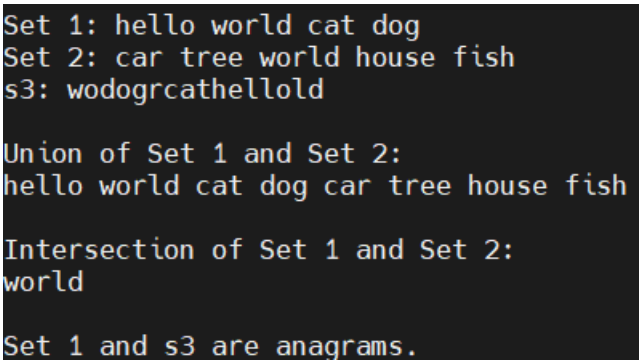
1. **StringSet StringSet::operator|(const StringSet &other)**
   → return the union of two StringSet objects
   Format: StringSet result = ss1 | ss2;
2. **StringSet StringSet::operators&(const StringSet &other)**
   → return the intersection of two StringSet objects
   Format: StringSet result = ss1 & ss2;
3. **bool StringSet::operator*(const string &str)**
   → Combine all the strings of current class StringSet, which is s1, and check whether s1 are an anagrams of s3
   Format: bool result = ss1 * “example”;
4. **void operator+=(StringSet &set, const string &str)**
   → add str string to the StringSet set
   Format: ss1 += “example”;
5. **ostream &operator<<(ostream &out, const StringSet &set)**
   → output all the strings in the set

Example

Use **cin** to input:

Set 1 = {”hello”, “world”, “cat”, “dog”, “hello”}
Set 2 = {“car”, “tree”, “world”, “house”, “fish”}
s3 = “wodogrcathellold”

Output:



Note

- The elements in the set **must not be repeated**.
  For example, in the illustration above, Set 1 cannot contain two "hello".
- The **union result has a specific order**: First, store Set 1 into the result, and then store the rest of the non-repeating part of Set 2 into result.
  For example, Set 1 = {"a", "b", "c" }, Set 2 = {"d", "b", "e", "a"}
  → union result should be {"a", "b", "c"} + {"d", ~~"b"~~, "e", ~~"a"~~} = {"a", "b", "c", "d", "e"}
- The **intersection result has a specific order**: Store the result according to the order of Set 1.
  For example, Set 1 = {"a", "b", "c" }, Set 2 = {"d", "b", "e", "a"}
  → intersection result should be {"a", "b"}
- You are **not allowed to use built-in sets** such as set or unordered_set to implement the function.
- **anagram** definition: a **word formed by rearranging the letters of another**.
  For example, "listen" and "silent" are anagrams of each other.

Compile

g++ main.cpp StringSet.cpp -o Mid04

Run

./Mid04

OJ

/home/share/demo_OOP112_2 Mid 04