1. **Collections Mystery.** Consider the following method:

```java
public Map<String, String> mystery(List<String> list1, List<String> list2) {
    Map<String, String> result = new TreeMap<String, String>();
    for (int i = 0; i < list1.size(); i++) {
        result.put(list1.get(i), list2.get(i));
        result.put(list2.get(i), list1.get(i));
    }
    return result;
}
```

The three entries below have specific values for the first and second parameters to method mystery (when you see text like the letter b, that indicates that the list stores the String "b" in that position). For each entry, indicate what values would be stored in the map returned by method mystery if the given lists are passed as parameters. Map elements should be listed with "key=value" elements, as in {b=z, d=e}. You should list the map entries in their proper order.

List: [b, l, u, e]   List: [s, p, o, t]

Map returned:

List: [k, e, e, p]   List: [s, a, f, e]

Map returned:

List: [s, o, b, e, r]   List: [b, o, o, k, s]

Map returned:
2. **Collections Mystery.** Consider the following method:

```java
public static Map<String, Integer> mystery1(List<String> list) {
    Map<String, Integer> result = new TreeMap<String, Integer>();
    for (int i = 0; i < list.size(); i++) {
        result.put(list.get(i), i);
    }
    return result;
}
```

The three entries below have specific values for the parameter to method mystery. For each entry, indicate what values would be stored in the Map returned by method mystery if the given maps are passed as parameters.

list: [four, score, and, seven, years, ago]
map returned: __________________________________________________________

list: [Fred, Wilma, Barney, Fred, Bart, Homer]
map returned: __________________________________________________________

list: [to, be, or, not, to, be, that, is, the, question]
map returned: __________________________________________________________