Chapter 17-2. Lab
Managing Springfield Residents using Iterators & Sets

Objective:

This lab assignment extends the "Springfield Residents" program by incorporating iterators for vector traversal, std::set for managing unique occupations, and exception handling using std::runtime_error to deal with invalid resident data.

Requirements:

1. **Review and Setup:**
    - Review the previous "Springfield Residents" lab. Ensure you have a working Resident class and basic vector operations (Figure 1 shows the class diagram).
2. **Resident Class Modifications:**
    - Modify the Resident class constructor to throw a std::runtime_error exception if the occupation is "Professional Nap Taker" or "Cloud Gazer".
3. **Iterators:**
    - Use iterators for traversing the residents vector in the following sections:
        - Displaying all residents.
        - Displaying residents over 30 years old.
        - Displaying residents who are students.
4. **Sets for Unique Occupations:**
    - Create a std::set to store the unique occupations of the residents.
    - Add occupations to the set as you add residents to the vector.
    - Implement a function to display all unique occupations from the set.
5. **Additional Vector Operations with Iterators:**
    - Implement the following operations using iterators:
        - **Insert:** Insert a new resident into the residents vector at a specific position.
        - **Delete:** Delete a resident from the residents vector based on their name.
6. **Sorting with Custom Comparator:**
    - Use std::sort with a custom comparator function or lambda expression to sort the residents by name (alphabetically).
    - Display the sorted list of residents.
7. **Exception Handling:**
    - In the main function, use a try-catch block to handle potential std::runtime_error exceptions when creating Resident objects.
    - Provide informative error messages to the user if an invalid occupation is detected.

8. **Unacceptable Occupations:**
    - The occupations "Professional Nap Taker" and "Cloud Gazer" should trigger an exception.
    - Add the following residents to the code, which should cause exceptions:
        - Resident("Barney Gumble", 38, "Professional Nap Taker")
        - Resident("Jimbo Jones", 11, "Cloud Gazer")

Modified Skeleton Code:

```
int main() {
    vector<Resident> residents;
    set<string>      occupations;

    residents.push_back(Resident("Homer Simpson", 39, "Nuclear Safety Inspector"));
    residents.push_back(Resident("Marge Simpson", 36, "Housewife"));
    residents.push_back(Resident("Bart Simpson", 10, "Student"));
    residents.push_back(Resident("Lisa Simpson", 8, "Student"));
    residents.push_back(Resident("Maggie Simpson", 1, "Baby"));

    // TODO - Add occupations to the set

    // TODO - Attempt to add residents with invalid occupations

    //TODO: Add safely two more residents to the vector using insert()
    // (e.g., "Ned Flanders", 60, "Store Owner")
    // (e.g., "Mr. Skinner", 64, "School Principal")

    //TODO: Display all residents using iterators

    //TODO: Display residents over 30 years old using iterators

    //TODO: Display residents who are students using iterators

    //TODO: Sort the residents by name and display them

    //TODO: Search for a resident by name (ask operator to enter a name)

    //TODO: Display unique occupations

    cout << "\nAll done!" << endl;


}
```

Figure 1. UML Class Diagram representation of the Resident class.

```
+-----------------------------------+
|   Resident                        |
+-----------------------------------+
| # name: string                    |
| # age: int                        |
| # occupation: string              |
+-----------------------------------+
| + Resident(name, age, occupation) |
| + getName(): string               |
| + getAge(): int                   |
| + getOccupation(): string         |
| + display(): void                 |
+-----------------------------------+
```

Figure 2. Meeting the people of Springfield

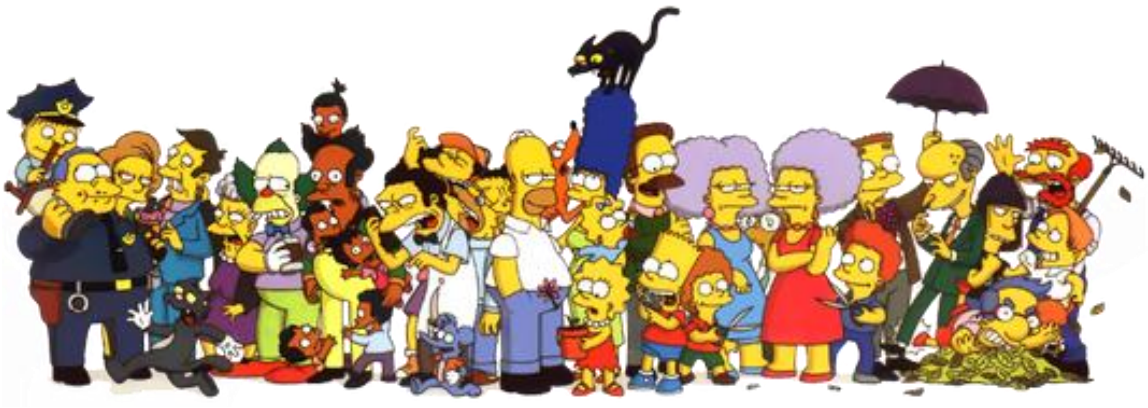

Figure 3. Expected output

```
Error: Could not add Barney Gumble. Invalid Occupation: Professional Nap Taker
Error: Could not add Jimbo Jones. Invalid Occupation: Cloud Gazer
All residents:
Name: Homer Simpson, Age: 39, Occupation: Nuclear Safety Inspector
Name: Marge Simpson, Age: 36, Occupation: Housewife
Name: Ned Flanders, Age: 60, Occupation: Store Owner
Name: Bart Simpson, Age: 10, Occupation: Student
Name: Lisa Simpson, Age: 8, Occupation: Student
Name: Maggie Simpson, Age: 1, Occupation: Baby
Name: Mr. Skinner, Age: 64, Occupation: School Principal

Residents over 30 years old:
Name: Homer Simpson, Age: 39, Occupation: Nuclear Safety Inspector
Name: Marge Simpson, Age: 36, Occupation: Housewife
Name: Ned Flanders, Age: 60, Occupation: Store Owner
Name: Mr. Skinner, Age: 64, Occupation: School Principal

Residents who are students:
Name: Bart Simpson, Age: 10, Occupation: Student
Name: Lisa Simpson, Age: 8, Occupation: Student

Sorted by name:
Name: Bart Simpson, Age: 10, Occupation: Student
Name: Homer Simpson, Age: 39, Occupation: Nuclear Safety Inspector
Name: Lisa Simpson, Age: 8, Occupation: Student
Name: Maggie Simpson, Age: 1, Occupation: Baby
Name: Marge Simpson, Age: 36, Occupation: Housewife
Name: Mr. Skinner, Age: 64, Occupation: School Principal
Name: Ned Flanders, Age: 60, Occupation: Store Owner

Search for a resident by name:
Enter resident name to search: Bart Simpson
Resident found:
Name: Bart Simpson, Age: 10, Occupation: Student

Unique Occupations:
Baby
Housewife
Nuclear Safety Inspector
School Principal
Store Owner
Student

All done!
```