【C++】Day38(2)

Class	C++
Date	@January 16, 2022
Material	
Series Number	
Summary	

【Ch10】Generic Algorithm

10.2.3 Algorithms That Reorder Container Elements

Some algorithms rearrange the order of elements within a container. An obvious example of such an algorithm is sort.

Eliminating Duplicates

To eliminate the duplicated words, we will first sort the vector so that duplicated words appear adjacent to each other.

Once the vector is sorted, we can use another library algorithm, named `unique`, to reorder the vector so that the unique elements appears in the first part of the vector.

```
void elimDuplicates(vector<string> &words) {
  //sort words alphabetically so we can find duplicates
  words.sort(words.begin(), words.end());
  //unique reorder the input range so that each words appear once in the front portion of the range
  //and returns an iterator one past the unique range.
  auto end_unique = unique(words.begin(), words.end());
  words.erase(end_unique, words.end());
}
```

`unique` does not remove any elements. Instead, it overwrites the adjacent duplicates so that the unique elements appear at the front of the sequence. The iterators returned by unique denotes one past the last unique element.

Note: The library algorithms operate on iterators, not containers. Therefore, an algorithm cannot add or remove elements.

Exercise

Exercise 10.9: Implement your own version of `elimDups`. Test your program by printing the `vector` after you read the input, after the call to `unique`, and after the call to `erase`.

See 10_9.cpp for code