Short Notes on List Functions

Function	Action	Result
push_front(5)	Add 5 to the front	5 2 4
push_back(7)	Add 7 to the end	5 2 4 7
insert(it, 3)	Insert 3 before it	5 3 2 4 7
insert(it, 2, 9)	Insert two 9s at it	5 9 9 2 4 7
erase(it)	Remove element pointed to by it	5 2 4 7
resize(6, 0)	Resize to 6 elements, fill with 0	5 2 4 7 0 0
remove(2)	Remove all 2s	5 4 7 0 0
remove_if(x > 3)	Remove elements greater than 3	3 0 0
sort()	Sort the list	0 0 3
unique()	Remove consecutive duplicates	0 3
reverse()	Reverse the list	3 0
merge(other_list)	Merge with {1, 2}	0 1 2 3
splice(it, other_list)	Insert elements of other_list at it	Inserted list elements at specified spot

---

Basic Operations

1. **push_front(value) / emplace_front(value)**: Insert at the beginning.
2. **push_back(value) / emplace_back(value)**: Insert at the end.
3. **pop_front() / pop_back()**: Remove from the front/back.

Access

4. **front()**: Access the first element.
5. **back()**: Access the last element.

Modifiers

6. **resize(n, value)**: Change the size of the list to n. New elements (if needed) are set to value.
7. **insert(pos, value)**: Insert value at the position pointed to by the iterator pos.
   - **insert(pos, n, value)**: Insert n copies of value at position pos.
8. **erase(pos)**: Remove the element at position pos.
   - **erase(start, end)**: Remove elements in the range [start, end).
9. **remove(value)**: Remove all occurrences of value.
10. **remove_if(condition)**: Remove elements based on a condition.
11. **clear()**: Remove all elements.
12. **swap(other_list)**: Swap the contents of the current list with other_list.

Traversal

13. Use **begin(), end(), rbegin(), rend()** for iterators.
14. Loop with **for (auto x : list)** for easy traversal.

Sorting and Unique

15. **sort()**: Sort elements in ascending order or using a comparator.
16. **unique()**: Remove consecutive duplicates.

Reversing

17. **reverse()**: Reverse the order of elements.

Merge

18. **merge(other_list)**: Combine two sorted lists into one sorted list.

Splice

19. **splice(pos, other_list)**: Transfer elements from another list to the current list at position pos.
20. **splice(pos, other_list, it)**: Transfer the element pointed to by it from other_list to pos.
21. **splice(pos, other_list, start, end)**: Transfer a range [start, end) from other_list to pos.

Size

22. **size()**: Get the number of elements in the list.
23. **empty()**: Check if the list is empty.
24. **max_size()**: Return the maximum number of elements the list can hold.