

std::set

09 June 2021 07:07

1. std::set– sorted value using binary search tree

```
std::set<int> first; // empty set of ints

int myints[] = {10,20,30,40,50};
std::set<int> second (myints,myints+5); // range
std::set<int> third (second); // a copy of second

std::set<int> fourth (second.begin(), second.end()); // iterator ctor

std::set<int, classcomp> fifth; // functor Compare
```

2. Second template parameter (Compare)

This shall be a function pointer or a function object.

From <<https://cplusplus.com/reference/set/set/set/>>

Uses the default std::less

3. Iterators – const, non-const bidirectional and reverse iterator

4. Members access – via itr

5. Size

- size - number of elements currently used
- empty

MAX_SIZE - ???

6. Set size related

- erase(it) or erase(itr1, itr2) or erase(value) - and reduce size
- clear() - make size 0
- std::pair<std::set<int>::iterator, bool> insert(value), insert(itr, value) or insert(itr1, itr2)
- std::pair<std::set<int>::iterator, bool> emplace(value)

7. Operator =, []

I

8. Special – count(key) returns 1 or 0, find(value) returns std::set<int>::iterator

I

9. F

I

10. F

11. F

I

12. F

I

13. F

I

- 14. F
I
- 15. F
I
- 16. F
I
- 17. D
I
- 18. F
I
- 19. F
I
- 20. F
I
- 21. F
- 22. F
I
- 23. F
I
- 24. F
I
- 25. F
I
- 26. F
I
- 27. F
I
- 28. D
I
- 29. F
I
- 30. F
I
- 31. F
I
- 32. F
- 33. F
I
- 34. F
I
- 35. F
I
- 36. F
I
- 37. F
I

I

38. F

I

39. D

I

40. F

I

41. F

I

42. F

I

43. F

44. F

I

45. F

I

46. F

I

47. F

I

48. F

I