

42 Lecture - CS304

Important Subjective

1. What is an iterator in C++?

Answer: An iterator in C++ is an object used to traverse through the elements of a container and perform operations on them.

What is the purpose of an iterator?

Answer: The purpose of an iterator is to provide a way to access the data stored in a container without exposing its internal representation, making the container more flexible and reusable.

What is the difference between an input iterator and an output iterator?

Answer: An input iterator is used to read data from a container, while an output iterator is used to write data to a container.

What is the difference between a forward iterator and a bidirectional iterator?

Answer: A forward iterator can only move forward in a container, while a bidirectional iterator can move both forward and backward.

What is the difference between a random access iterator and a bidirectional iterator?

Answer: A random access iterator provides constant time access to any element in a container, while a bidirectional iterator only provides constant time access to the next or previous element.

How is the complexity of an iterator defined?

Answer: The complexity of an iterator is defined by the amount of time it takes to perform certain operations, such as incrementing or decrementing the iterator.

What is the difference between a container and an iterator?

Answer: A container is a data structure that stores elements, while an iterator is an object used to traverse through the elements of a container.

What is the purpose of the `std::begin()` and `std::end()` functions?

Answer: The `std::begin()` function returns an iterator pointing to the first element in a container, while the `std::end()` function returns an iterator pointing to the end of the container.

What is the difference between a constant iterator and a regular iterator?

Answer: A constant iterator is used to iterate through a container without allowing modifications to the elements, while a regular iterator allows modifications to the elements.

How are iterators used in algorithms from the Standard Template Library?

Answer: Iterators are used as arguments to algorithms in the Standard Template Library to specify which elements in a container to operate on.