

```

#include "FibonacciSequenceIterator.h"
#include <cassert>

FibonacciSequenceIterator::FibonacciSequenceIterator(const FibonacciSequenceGenerator&
aSequenceObject, long long aStart) noexcept
    : fSequenceObject(aSequenceObject), fIndex(aStart - 1) {}

const long long& FibonacciSequenceIterator::operator*() const noexcept {
    return *fSequenceObject;
}

FibonacciSequenceIterator& FibonacciSequenceIterator::operator++() noexcept {
    ++fIndex;
    fSequenceObject.next(); // Advance the sequence generator
    return *this;
}

FibonacciSequenceIterator FibonacciSequenceIterator::operator++(int) noexcept {
    FibonacciSequenceIterator temp = *this;
    ++(*this);
    return temp;
}

bool FibonacciSequenceIterator::operator==(const FibonacciSequenceIterator& aOther) const
noexcept {
    return fSequenceObject.id() == aOther.fSequenceObject.id() && fIndex == aOther.fIndex;
}

bool FibonacciSequenceIterator::operator!=(const FibonacciSequenceIterator& aOther) const
noexcept {
    return !(*this == aOther);
}

FibonacciSequenceIterator FibonacciSequenceIterator::begin() const noexcept {
    return FibonacciSequenceIterator(fSequenceObject, 1);
}

FibonacciSequenceIterator FibonacciSequenceIterator::end() const noexcept {
    return FibonacciSequenceIterator(fSequenceObject, 93); // The end position is at Fibonacci
number 93
}

```