```cpp
//
//  FibonacciSequenceIterator.cpp
//  problemset2
//
//  Created by Xinzhe Yu on 7/4/2025.
//
#include <iostream>
#include "FibonacciSequenceIterator.h"

FibonacciSequenceIterator::FibonacciSequenceIterator(FibonacciSequence*
 aSequence, uint64_t aStart) noexcept:
    fSequence(aSequence),
    fIndex(aStart)
{
//    if the aSequence is existing, let the aSquence iterate to the
 position of fIndex, if no fIndex, it will start from 0
    if (fSequence != nullptr)
    {
        fSequence->begin();
        for (uint64_t i = 1; i < fIndex; ++i)
        {
            ++(*fSequence);
        }
    }
}

const uint64_t& FibonacciSequenceIterator::operator*() const noexcept{
    return **fSequence;
}

FibonacciSequenceIterator& FibonacciSequenceIterator::operator++() noexcept{
        ++(*fSequence);
        ++fIndex;
    return *this;
}

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

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

FibonacciSequenceIterator FibonacciSequenceIterator::begin() const noexcept
{
    return FibonacciSequenceIterator(fSequence, 0);
}
```

```
FibonacciSequenceIterator FibonacciSequenceIterator::end() const noexcept
{
    return FibonacciSequenceIterator(fSequence, MAX_FIBONACCI);
}
```