

```
1 #include "FibonacciSequenceGenerator.h"
2
3 // Constructor to set up a Fibonacci sequence
4 FibonacciSequenceGenerator::FibonacciSequenceGenerator(const      ↗
    std::string& aID) noexcept
5     : fID(aID), fPrevious(0), fCurrent(1) {}
6
7 // Get sequence ID
8 const std::string& FibonacciSequenceGenerator::id() const noexcept {
9     return fID;
10 }
11
12 // Get current Fibonacci number
13 const long long& FibonacciSequenceGenerator::operator*() const noexcept ↗
    {
14     return fCurrent;
15 }
16
17 // Type conversion to bool
18 // Returns true if there is a next Fibonacci number to be generated
19 FibonacciSequenceGenerator::operator bool() const noexcept {
20     return hasNext();
21 }
22
23 // Reset sequence generator to first Fibonacci number
24 void FibonacciSequenceGenerator::reset() noexcept {
25     fPrevious = 0;
26     fCurrent = 1;
27 }
28
29 bool FibonacciSequenceGenerator::hasNext() const noexcept {
30     return fCurrent <= (LLONG_MAX - fPrevious);
31 }
32
33 // Advance to next Fibonacci number
34 void FibonacciSequenceGenerator::next() noexcept {
35
36     long long lNext = fCurrent + fPrevious;
37     fPrevious = fCurrent;
38     fCurrent = lNext;
39 }
40
41
42
```