

```
1 #include "KeyProvider.h"
2 #include <cctype>
3 #include <cassert>
4
5 std::string KeyProvider::preprocessString(const std::string& aString) ➤
    noexcept {
6     std::string result;
7     for (char c : aString)
8     {
9         if (isalpha(c))
10        {
11            result += toupper(c);
12        }
13    }
14    return result;
15 }
16
17 KeyProvider::KeyProvider(const std::string& aKeyword, const std::string& ➤
    aSource) noexcept {
18     std::string lKeyword = preprocessString(aKeyword);
19     std::string lSource = preprocessString(aSource);
20     for (size_t i = 0; i < lSource.length(); i++)
21     {
22         fKeys += lKeyword[i % lKeyword.length()];
23     }
24     fIndex = 0;
25     assert(fKeys.length() == lSource.length());
26 }
27
28 char KeyProvider::operator*() const noexcept {
29     return fKeys[fIndex];
30 }
31
32 KeyProvider& KeyProvider::operator++() noexcept {
33     fIndex++;
34     return *this;
35 }
36
37 KeyProvider KeyProvider::operator++(int) noexcept {
38     KeyProvider old = *this;
39     ++(*this);
40     return old;
41 }
42
43 bool KeyProvider::operator==(const KeyProvider& aOther) const noexcept {
44     return fIndex == aOther.fIndex && fKeys == aOther.fKeys;
45 }
46
47 bool KeyProvider::operator!=(const KeyProvider& aOther) const noexcept {
48     return !(*this == aOther);
49 }
50
51 KeyProvider KeyProvider::begin() const noexcept {
```

```
52     KeyProvider temp = *this;
53     temp.fIndex = 0;
54     return temp;
55 }
56
57 KeyProvider KeyProvider::end() const noexcept {
58     KeyProvider temp = *this;
59     temp.fIndex = fKeys.size();
60     return temp;
61 }
62
```