```
2 //Sartaj Khan
 3 //Date: 26/04/23
 5 #include "LZWCompressor.h"
 7
 8
   bool LZWCompressor::readK() noexcept {
 9
        if (fInput[fIndex + 1]) {
            fK = fInput[++fIndex];
10
11
            return true;
12
        }
        fK = -1;
13
        return false;
14
15 }
16
17
18 void LZWCompressor::start() {
19
        fTable = LZWTable();
        fTable.initialize();
20
21
        fK = fInput[0];
        fW = fTable.lookupStart(fK);
22
23
        fCurrentCode = nextCode();
24 }
25
26
   uint16_t LZWCompressor::nextCode() {
28
        if (fK == -1) {
29
            return -1;
30
31
        while (readK()) {
32
            PrefixString next = fW + fK;
33
            if (fTable.contains(next)) {
34
                fW = next;
35
            }
            else {
36
37
                fTable.add(next);
38
                uint16_t code = fW.getCode();
                fW = fTable.lookupStart(fK);
39
40
                return code;
41
            }
42
43
        return fW.getCode();
44 }
45
46
   LZWCompressor::LZWCompressor(const std::string& aInput) :
48
        fInput(aInput),
        fIndex(0)
49
```

```
...winburne University\DSP\Midterm\Midterm\LZWCompressor.cpp
```

```
2
```

```
50 {
51
       start();
52 }
53
54 const uint16_t& LZWCompressor::operator*() const noexcept {
        return fCurrentCode;
56 }
57
58 LZWCompressor& LZWCompressor::operator++() noexcept {
59
       fCurrentCode = nextCode();
60
       return *this;
61 }
62
63 LZWCompressor LZWCompressor::operator++(int) noexcept {
64
       LZWCompressor old = *this;
       ++(*this);
65
66
       return old;
67 }
68
69 bool LZWCompressor::operator==(const LZWCompressor& a0ther) const noexcept {
       return (fInput == a0ther.fInput) && (fIndex == a0ther.fIndex && fK ==
70
         aOther.fK && fCurrentCode == aOther.fCurrentCode);
71 }
72
73 bool LZWCompressor::operator!=(const LZWCompressor& a0ther) const noexcept {
74
       return !(*this == a0ther);
75 }
76
77 LZWCompressor LZWCompressor::begin() const noexcept {
78
       LZWCompressor begin = *this;
79
       begin.fIndex = 0;
80
       begin.start();
81
       return begin;
82 }
83
84 LZWCompressor LZWCompressor::end() const noexcept {
85
       LZWCompressor end = *this;
86
       end.fIndex = end.fInput.size() - 1;
       end.fCurrentCode = -1;
87
88
       end.fK = -1;
89
       return end;
90 }
91
```