```
01 //
02 // Created by along on 17-11-11.
03 //
04
05 #include <vector>
06 #include <cstring>
07 #include <fstream>
08 #include <sstream>
09 #include "BitTree.h"
10
11 using namespace std;
12
13 int main(int argv, char **argc) {
14     ifstream fileIn;
15     ofstream dotFile("tree.dot");
16     vector<int> front;
17     vector<int> middle;
18     BitTree<int> tree;
19
20     if (argv > 1) {
21         for (int i = 0; i != argv; ++i) {
22             if (0 == strncmp(argc[++i], "-I", 2)) {
23                 int data;
24                 string line;
25                 istringstream lineStream;
26                 fileIn.open(argc[++i]);
27                 getline(fileIn, line);
28                 lineStream.str(line);
29                 lineStream.clear();
30                 while (lineStream >> data) {
31                     front.push_back(data);
32                 }
33                 getline(fileIn, line);
34                 lineStream.str(line);
35                 lineStream.clear();
36                 while (lineStream >> data) {
37                     middle.push_back(data);
```

```
38                  }
39                  tree = BitTree<int>::CreateBitTree(front, ↩
40                  middle);
41              }
42          }
43      } else {
44          tree =
45              BitTree<int>::CreateBitTree({1, 2, 3, 4, 5, 6, 7,↩
46              8, 9, 0, 10, 0, 0, 0, 0, 0, 0, 0, 0, 11, 12}, ↩
47              true);
48      }
49
50      cout << " 先序非递归:";
51      tree.visitTree(tree.DLR);
52      cout << endl;
53      cout << " 中序非递归:";
54      tree.visitTree(tree.LDR);
55      cout << endl;
56      cout << " 后序非递归:";
57      tree.visitTree(tree.LRD);
58      cout << endl;
59
60      cout << " 先序递归:";
61      tree.visitTreeR(tree.DLR);
62      cout << endl;
63      cout << " 中序递归:";
64      tree.visitTreeR(tree.LDR);
65      cout << endl;
66      cout << " 后序递归:";
67      tree.visitTreeR(tree.LRD);
68      cout << endl;
69
70      BitTree<int>::BitTreeThreading(tree);
71      tree.printGraph(dotFile);
72
73      cout << " 先序线索:";
74      tree.threadVisit(tree.DLR);
```

```
75      cout << endl;
76      cout << " 中序线索:";
77      tree.threadVisit(tree.LDR);
78      cout << endl;
79      cout << " 后序线索:";
80      tree.threadVisit(tree.LRD);
81      cout << endl;
82
83      cout << " 层序非递归:";
84      tree.visitTreeCover();
85      cout << endl;
86
87      fileIn.close();
88      dotFile.close();
89      return 0;
90  }
91
```