

In-class notes: Implementing a graph

CS 14 - Data Structures

May 22, 2013

```
1  #include <map>
2  #include <string>
3  #include <list>
4
5  class Graph
6  {
7  public:
8      typedef char NodeID;
9
10 private:
11     struct Node
12     {
13         NodeID id;
14         std::string label;
15         double weight;
16
17         std::list<Node*> neighbors;
18     };
19
20     std::map<NodeID,Node*> id_index;
21
22 public:
23     void addVertex(NodeID u, std::string label, double weight)
24     {
25         Node *newnode = new Node();
26         newnode->id=u;
27         newnode->label=label;
28         newnode->weight=weight;
29         id_index.insert(std::pair<NodeID,Node*>(u,newnode));
30     }
31
32     void addEdge(NodeID u, NodeID v)
33     {
34         id_index.at(u)->neighbors.push_back(id_index.at(v));
35         id_index.at(v)->neighbors.push_back(id_index.at(u));
36     }
37
38
```

```

39     void removeVertex()
40     {
41         // TODO: implement
42     }
43
44     void removeEdge(NodeID u, NodeID v)
45     {
46         // remove v from u
47         for ( std::list<Node*>::iterator i = id_index.at(u)->neighbors.begin()
48             ; i != id_index.at(u)->neighbors.end()
49             ; i++) {
50             if ((*i)->id==v) {
51                 id_index.at(u)->neighbors.erase(i);
52                 break;
53             }
54         }
55
56         // remove u from v
57         for ( std::list<Node*>::iterator i = id_index.at(v)->neighbors.begin()
58             ; i != id_index.at(v)->neighbors.end()
59             ; i++) {
60             if ((*i)->id==u) {
61                 id_index.at(v)->neighbors.erase(i);
62                 break;
63             }
64         }
65     }
66
67     bool adjacent(NodeID u, NodeID v)
68     {
69         for ( std::list<Node*>::iterator i = id_index.at(u)->neighbors.begin()
70             ; i != id_index.at(u)->neighbors.end()
71             ; i++) {
72             if ((*i)->id==v) {
73                 return true;
74             }
75         }
76     }
77 };

```