```cpp
#include <iostream>
#include <string>
#include <fstream>
#include <iomanip>

using namespace std;


struct listtype {
string last;
string first;
int id;
listtype *link; };

void PrintTopTitles(ofstream &outf) {
outf << setw(13) << setfill(' ') << right << " " << "CSCI 112" << endl;
outf << setw(12) << setfill (' ') << right << " " << "Class List" <<endl;
outf << setw(35) << setfill('-') << right << "- " << endl;
outf << setw(6) << setfill(' ') << left << "ID" << " " << "Name" << endl;
outf << setw(35) << setfill('-') << right << "- " << endl;
}

void PrintBottomTitles(ofstream &outf) {
outf << setw(35) << setfill('-') << right << "- " << endl;
}

void CreateList(listtype *&head, listtype *&tail) {
    head = new listtype;
    tail = new listtype;
    head -> last = "  AAAA";
    tail -> last = "zzzzzz";
    head -> link = tail;
    tail -> link = NULL;
}

bool EmptyList(listtype *head, listtype *tail) {
    return (head -> link == tail);
}

void InsertList(listtype *head, listtype *tail, string last, string first, int id) {
    listtype *before, *after, *insert;
    insert = new listtype;
    insert -> last = last;
    insert -> first = first;
    insert -> id = id;
    before = head;
    after = head -> link;
    while (after != tail && insert->last > after->last) {
        before = after;
        after = after -> link; }
    before -> link = insert;
    insert -> link = after;
}

```

```
55  void ReadData(listtype *head, listtype *tail) {
56  int id;
57  string last, first;
58  ifstream inf;
59  inf.open("program2datafile.txt");
60  int position = -1;
61      while (!inf.eof()) {
62          inf >> last >> first >> id >> ws;
63          position = last.find(",");
64          if (position >=0) last.erase(position,1);
65          InsertList(head, tail, last, first, id);
66      }
67  }
68
69  void TraverseList(listtype *head, listtype *tail, ofstream &outf) {
70      listtype *c;
71      string temp;
72      if (!EmptyList(head, tail)) {
73          PrintTopTitles(outf);
74          c = head -> link;
75          while (c!= tail) {
76              temp = c -> first + " " + c -> last;
77              outf << setw(6) << setfill(' ') << left << c->id << " " << temp << endl;
78              c = c-> link;
79          }
80      PrintBottomTitles(outf);
81      }
82      else outf << "The List Is Empty" << endl;
83  }
84
85  void DeletefromList(listtype *head, listtype *tail, string last, string first, int
    id, ofstream &outf){
86      listtype *before, *next, *c;
87      before = head;
88      c = head -> link;
89      next = c-> link;
90      while (c!=tail && c->last < last) {
91          before = c;
92          c = next;
93          next = next -> link; }
94      if (c!=tail && c->last == last) {
95          before -> link = next;
96          delete c; }
97      else {
98          outf << "Element not in list" ;
99          }
100 }
101
102 void PrintSpaces(ofstream &outf) {
103     outf << endl << endl << endl << endl;
104 }
105
106 int main(){
107 ofstream outf;
```

```
108   outf.open("program2outputfile.txt");
109   listtype *head, *tail;
110   CreateList(head,tail);
111   ReadData(head, tail);
112   TraverseList(head,tail,outf);
113   PrintSpaces(outf);
114   InsertList(head, tail, "Skywalker", "Luke", 444);
115   TraverseList(head,tail,outf);
116   outf << "Luke Skywalker, ID #444, was added to the class." << endl;
117   PrintSpaces(outf);
118   DeletefromList(head, tail, "Organa", "Leia", 355, outf);
119   TraverseList(head, tail, outf);
120   outf << "Leia Organa, ID #355, was deleted from the class." << endl;
121   outf << endl;
122   system("pause");
123   }
124
```