NC State University
ECE Department ECE 720 W. Rhett Davis

Homework #4 Solution

Problem 1)

Here are the contents of the modified instcount.cpp file. The lines in red are the modified ones.

```
int sc_main(int argc, char* argv[])
{

    map<string,module*> mods;
    string line,first,second,current_module;
    size_t pos;
    ifstream f("LMS_pipe.hier");


    while (f.good()) {
      getline(f,line);
      pos=line.find(' ');
      first = line.substr(0,pos);
      second = line.substr(pos+1);
      //cout << "\"" << first << "\"" << endl;
      //cout << "\"" << second << "\"" << endl;
      if (first == "module") {
        current_module = second;
        mods[current_module]=new module(current_module);
        //cout << "module " << second << endl;
      }
      else if (second != "") {
        mods[current_module]->addInstance(first);
      }

    }
    f.close();

    cout << mods[current_module]->countInstances(mods)
         << " total instances" << endl;

    //map<string,module*>::iterator it;
    //for (it=mods.begin(); it != mods.end(); it++) {
    //  cout << "module " << it->second->name << endl;
    //  it->second->print();
    //  }

    return 0;
}
```

Here is a portion of the modified module.h file. A new method called countInstances was added.

```
class module {
public:
    string name;
    vector<string> instances;
    module (string n) { name = n; }
    void addInstance(string modname) { instances.push_back(modname); }
    void print();
    int countInstances(map<string,module*> &mods);
};
```

Here is the definition of the countInstances() method in the module.cpp file.

```
int module::countInstances(map<string,module*> &mods) {
  int count = 0;
  vector<string>::iterator it;
  for (it=instances.begin(); it < instances.end(); it++) {
    //cout << '"' << *it << "\" " << mods[*it] << endl;
    if (mods[*it] != 0)
      count+= mods[*it]->countInstances(mods);
    else
      count++;
  }
  return count;
}
```

When running this script, 10581 Total instances should be reported for the LMS_pipe.hier file.

Problem 2)

The final program should print the following output:

```
back
dog's
lazy
the
over
jumpped
fox
brown
quick
The
```

The contents of the **reader.h** file are given below:

```
#pragma once

#include <list>
#include <string>

using namespace std;

class reader {
  list<string> words;
public:
  reader( string filename );
  void reversePrint();
};
```

The contents of the **reader.cpp** file are given below:

```
#include "reader.h"
#include <iostream>
#include <fstream>

using namespace std;

reader::reader(string filename) {
  string line;
  size_t pos;
  ifstream f(filename.c_str());

  while (f.good()) {
    getline(f,line);
    while (line.length() > 0) {
      //cout << '"' << line << '"' << endl;
      if (line[0] == ' ')
        line=line.substr(1);
      else {
        pos=line.find(' ');
        if (pos != line.npos) {
          words.push_back(line.substr(0,pos));
          line=line.substr(pos+1);
      }
        else {
          words.push_back(line);
        line="";
      }
      }
    }
  }
  f.close();
}

void reader::reversePrint() {
  words.reverse();
  list<string>::iterator it;
  for (it=words.begin(); it != words.end(); it++) {
    cout << *it << endl;
    }
}
```