```
1: // $Id: testxless.cpp,v 1.22 2015-02-06 16:22:06-08 - - $
 3: // Example of how to use xless object on a pair of strings.
 4:
 5: #include <iomanip>
 6: #include <iostream>
7: #include <string>
8:
9: using namespace std;
10:
11: template <typename Type>
12: struct xless {
13:
       bool operator() (const Type &left, const Type &right) const {
14:
          return left < right;
15:
       }
16: };
17:
18: template <typename Type, class Less=xless<Type>>
19: void compare (const Type &left, const Type &right) {
20:
       Less less;
21:
       cout << boolalpha;</pre>
       bool is_less = less (left, right);
22:
23:
       bool is_equal = not less (left, right) and not less (right, left);
      bool is_greater = not is_less and not is_equal;
24:
       cout << left << " < right << " = " << is_less << endl;
25:
       cout << left << " == " << right << " = " << is_equal << endl;
26:
       cout << left << " > " << right << " = " << is_greater << endl;
27:
28:
       cout << endl;</pre>
29: }
30:
31: int main () {
       compare (string ("hello"), string ("world"));
32:
       compare (string ("qwert"), string ("qwert"));
33:
34:
       compare (string ("zxcvb"), string ("asdfg"));
35:
       return 0;
36: }
37:
38: //TEST// testxless >testxless.out 2>&1
39: //TEST// mkpspdf testxless.ps testxless.cpp* testxless.out
40:
```

\$cmps109-wm/Assignments/asg3-listmap-templates/misc/ 04/23/15 1/1 16:20:26 testxless.cpp.log 1: @@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting testxless.cpp 2: testxless.cpp: \$Id: testxless.cpp,v 1.22 2015-02-06 16:22:06-08 - - \$ 4: g++ -g -00 -Wall -Wextra -rdynamic -std=gnu++11 testxless.cpp -o testxle ss -lglut -lGLU -lGL -lX11 -lrt -lm 5: rm -f testxless.o 

\$cmps109-wm/Assignments/asg3-listmap-templates/misc/ testxless.out

```
1: hello < world = true
2: hello == world = false
3: hello > world = false
4:
5: qwert < qwert = false
6: qwert == qwert = true
7: qwert > qwert = false
8:
9: zxcvb < asdfg = false
10: zxcvb == asdfg = false
11: zxcvb > asdfg = true
12:
```