

```
1: // $Id: mutexclick.cpp,v 1.5 2014-05-30 22:46:35-07 - - $
2:
3: // Hello to threads.
4:
5: #include <iostream>
6: #include <mutex>
7: #include <thread>
8: #include <vector>
9: using namespace std;
10:
11: constexpr size_t LOOPS = 1000000;
12:
13: class counter {
14:     private:
15:         size_t count {0};
16:         mutex lock;
17:     public:
18:         counter& operator++() {
19:             lock.lock();
20:             ++count;
21:             lock.unlock();
22:             return *this;
23:         }
24:         friend ostream& operator<< (ostream&, const counter&);
25: };
26:
27: ostream& operator<< (ostream& out, const counter& ctr) {
28:     out << ctr.count;
29:     return out;
30: }
31:
32: void racer (size_t id, counter* count) {
33:     cout << "racer " << id << " looping " << LOOPS << " times" << endl;
34:     for (size_t i = 0; i < LOOPS; ++i) ++*count;
35:     cout << "racer " << id << " finished counter " << *count << endl;
36: }
37:
38: int main () {
39:     counter count;
40:     cout << "main starting" << endl;
41:     vector<thread> vecthr;
42:     for (size_t i = 0; i < 4; ++i) {
43:         vecthr.push_back (thread (racer, i, &count));
44:     }
45:     for (auto& t: vecthr) t.join();
46:     cout << "main finished counter " << count << endl;
47:     return 0;
48: }
49:
50: //TEST// alias TIME='/usr/bin/time -f "%E elapsed, %S kernel, %U user"'
51: //TEST// for i in 1 2 3 4
52: //TEST// do
53: //TEST//     TIME mutexclick >mutexclick.out$i 2>&1
54: //TEST// done
55: //TEST// more mutexclick.out* >mutexclick.lis </dev/null
56: //TEST// mkpspdf mutexclick.ps mutexclick.cpp* mutexclick.lis
57:
```

```
t -lGLU -lGL -lX11 -lm -lrt
```

```
1: ::::::::::::::
2: mutexclick.out1
3: ::::::::::::::
4: main starting
5: racer 0 looping 1000000 times
6: racer 1 looping 1000000 times
7: racer racer 23 looping 1000000 looping 1000000 times
8: times
9: racer 1 finished counter 3527625
10: racer 3 finished counter 3557958
11: racer 0 finished counter 3968156
12: racer 2 finished counter 4000000
13: main finished counter 4000000
14: 0:01.28 elapsed, 3.34 kernel, 1.24 user
15: ::::::::::::::
16: mutexclick.out2
17: ::::::::::::::
18: main starting
19: racer 0 looping 1000000 times
20: racer 1 looping 1000000 times
21: racer 2 looping 1000000 times
22: racer 3 looping 1000000 times
23: racer 1 finished counter 3851827
24: racer 2 finished counter 3903853
25: racer 3 finished counter 3944624
26: racer 0 finished counter 4000000
27: main finished counter 4000000
28: 0:01.58 elapsed, 4.29 kernel, 1.43 user
29: ::::::::::::::
30: mutexclick.out3
31: ::::::::::::::
32: main starting
33: racer 0 looping 1000000 times
34: racer 1 looping 1000000 times
35: racer 2 looping 1000000 times
36: racer 3 looping 1000000 times
37: racer 1 finished counter 3829866
38: racer 0 finished counter 3977342
39: racer 2 finished counter 3983534
40: racer 3 finished counter 4000000
41: main finished counter 4000000
42: 0:02.56 elapsed, 7.30 kernel, 1.90 user
43: ::::::::::::::
44: mutexclick.out4
45: ::::::::::::::
46: main starting
47: racer 0 looping 1000000 times
48: racer 1 looping 1000000 times
49: racer 2 looping 1000000 times
50: racer 3 looping 1000000 times
51: racer 1 finished counter 3569648
52: racer 2 finished counter 3880671
53: racer 3 finished counter 3919142
54: racer 0 finished counter 4000000
55: main finished counter 4000000
56: 0:01.61 elapsed, 4.37 kernel, 1.48 user
```