```
1   template <class T>
2   bool multiCompareAndSet(std::atomic<T*> *target,
3                                T *expect,
4                                T *update,
5                                int count) {
6     atomic {
7       for (int i = 0; i < count; i++) {
8         if (*target[i] != expected[i])
9           return false;
10      }
11      for (int i = 0; i < count; i++)
12        *target[i] = update[i];
13      return true;
14    }
15  }
```

FIGURE 20.3 Pseudocode for multiCompareAndSet(). This code should execute atomically.