My first run of inspect can be seen below. If fails! Oh noes. Upon inspecting the trace given to us by the program we see that there's a small inconsistency with the pthread_cond code. The producer is using a while (correct usage of pthread_cond_wait), but the consumer is using an if. What happens is that one consumer gets through and can flag a producer. Then the second can get through at the same time. This is because it doesn't execute the pthread_cond_wait again. If it had it would be stopped dead in its tracks. Once it gets through it can consume something that doesn't exist and cause the assertion to happen.

By changing the if to a while the problem is resolved.

Inspect before fix:

```plaintext
=== run 1 ===
0 produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1

=== run 2 ===
0 produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
<<
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1

=== run 3 ===
0 produces 0, item, qsize = 0
<<
0 consumed 0, item, qsize = 1
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1

=== run 4 ===
0 produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
<<
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1

=== run 5 ===
0 produces 0, item, qsize = 0
<<
0 consumed 0, item, qsize = 1
0 produces 1, item, qsize = 0
0 consumed 0, item, qsize = 1

=== run 6 ===
0 produces 0, item, qsize = 0
<<
0 consumed 0, item, qsize = 1
0 produces 1, item, qsize = 0
0 consumed 0, item, qsize = 1
<<
=== run 7 ===
0 produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
```
<<
0  produces 1, item, qsize = 0
0 consumed 0, item, qsize = 1
=== run 8 ===
0  produces 0, item, qsize = 0
<<
0 consumed 0, item, qsize = 1
0 produces 1, item, qsize = 0
0 consumed 0, item, qsize = 1
=== run 9 ===
0  produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
<<
=== run 10 ===
0  produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
0 produces 1, item, qsize = 0
0 consumed 0, item, qsize = 1
=== run 11 ===
<<
1  produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1
=== run 12 ===
0  produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
<<
1  produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1
=== run 13 ===
0  produces 0, item, qsize = 0
<<
0 consumed 0, item, qsize = 1
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1
=== run 14 ===
0  produces 0, item, qsize = 0
0 consumed 0, item, qsize = 1
<<
1 produces 1, item, qsize = 0
1 consumed 0, item, qsize = 1
=== run 15 ===
0  produces 0, item, qsize = 0
<<
0 consumed 0, item, qsize = 1
0 consumed 0, item, qsize = 0
prodcons.c:27:producer:Assertion 'qsize >= 0' failed.
Found an assertion violation in the test target. Stopping Search...

Total number of runs: 15, killed-in-the-middle runs: 2
Transitions explored: 508
Used time (seconds): 0.754654