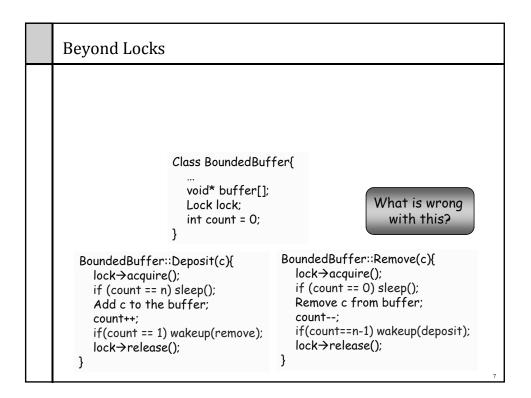
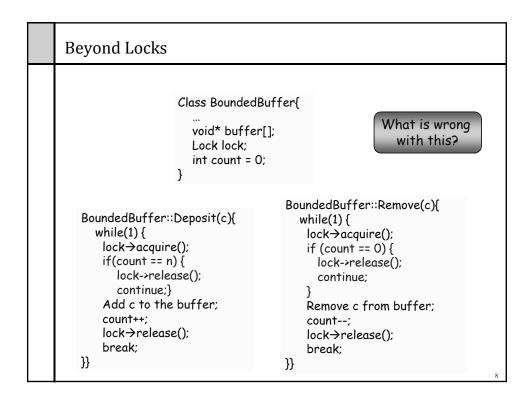
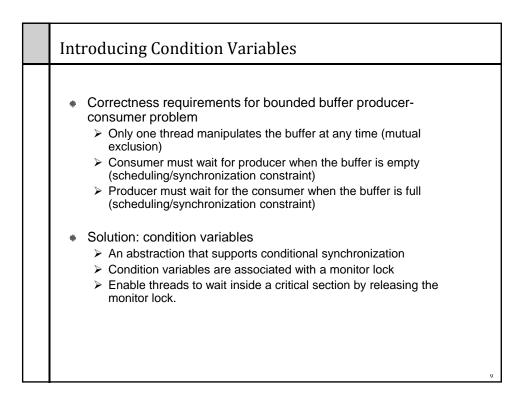
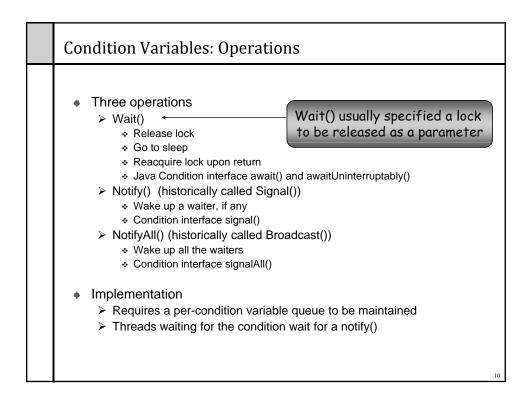


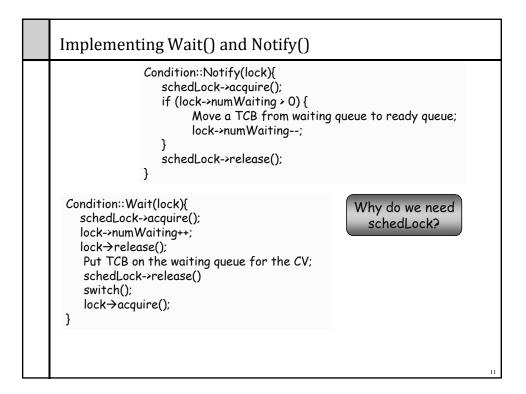
| Beyond Locks   |   |
|--|---|
| Class BoundedBuf<br><br>void* buffer[];<br>Lock lock;<br>int count = 0;<br>}<br>BoundedBuffer::Deposit(c){<br>if (count == n) glagn();                             |   |
| <pre>if (count == n) sleep();<br/>lock-&gt;acquire();<br/>Add c to the buffer;<br/>count++;<br/>lock-&gt;release();<br/>if(count == 1) wakeup(remove);<br/>}</pre> | lock->acquire();<br>Remove c from buffer;<br>count;<br>lock->release();<br>if(count==n-1) wakeup(deposit);<br>} |
| -  | 6   |

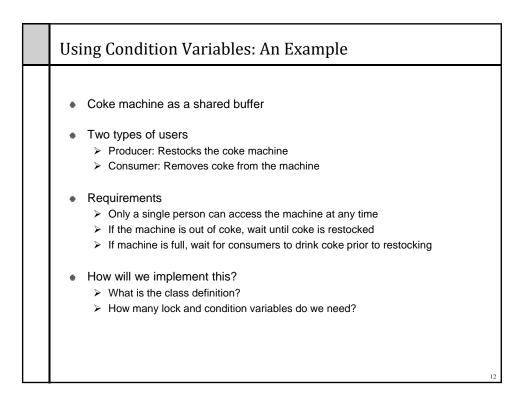












| Coke Machine Example   |   |  |
|--|---|--|
| Class Coke,<br><br>Lock loc<br>int coun<br>Conditio<br>}   | k;  |  |
| CokeMachine::Deposit(){<br>lock→acquire();<br>while (count == n) {<br>notFull.wait(&lock); }<br>Add coke to the machine;<br>count++;<br>notEmpty.notify();<br>lock→release();<br>} | CokeMachine::Remove(){<br>lock→acquire();<br>while (count == 0) { |  |

|  | Class Coke  | eMachine{  |
|--|---|--|
| Liveness<br>issue  | <br>Lock lock;<br>int count = 0;<br>Condition notFull, notEmpty;<br>} |  |
| CokeMachine::D<br>lock→acquire<br>while (count<br>notFull<br>Add coke to<br>count++;<br>notEmpty.not<br>lock→release | ;();<br>== n) {<br>.wait(&lock); }<br>the machine;<br>tify();         | CokeMachine::Remove(){<br>lock→acquire();<br>while (count == 0) {<br>notEmpty.wait(&lock); }<br>Remove coke from to the machine;<br>count;<br>lock→release();<br>notFull.notify(); |

