Proving impossibility

- We saw there was a 2-state DFA for the formula:

$$\forall x \left(Q_a(x) \Rightarrow \exists y \left(y > x \ \& Q_b(y) \right) \right)$$

- If "∃!" means "there exists unique", is there a way to express this in the logic?

$$\forall x \left(Q_a(x) \Rightarrow \exists!\, y \left(y > x \ \& Q_b(y) \right) \right)$$

- *Related question*: What does this code do on a stream of *a*'s and *b*'s?

```
bool check(stream s) {
   int count = 0;
   while (!s.empty()) {
      if (s.get() == 'a')
         count++;
      else // s.get() == 'b'
         if (count > 0)
            count--;
   }
   return count == 0;
}
```

Formal argument

- **Claim**: The strings satisfying $\forall x \left(Q_a(x) \Rightarrow \exists!\, y(y > x \;\& Q_b(y)) \right)$ define a non-regular language L [Then we're done. Why?]
  - **Sub-claim**: $\forall n \geq 0,\ a^n b^n \in L$ [Proof?]

- **Proof** [Pumping lemma/proof by contradiction]:
  - Suppose an n state DFA accepts L
  - The states reached on $a^0, a^1, \ldots, a^n$ cannot all be distinct [why?]
  - Hence, there is a repeated state and for infinitely many $m > n,\ a^m b^n \in L$

    CONTRADICTION

Undecidability

- **Fact 1**: The number of strings are over the alphabet {0, 1} is countably infinite
- **Fact 2**: The number of languages (language = set of strings) over the alphabet {0, 1} is uncountable
  - Proof similar to Cantor's diagonalization

- **Consequence**: There is no algorithm to decide membership in some languages