

# CS313K: Logic, Sets, and Functions

J Strother Moore  
Department of Computer Sciences  
University of Texas at Austin

(Lecture 7)

# Announcements

If you don't *understand* your Quiz and homework answers, you won't pass the exams!

Learn to *prettyprint* your definitions and formulas.

# Prettyprinting

```
(implies (and (consp x) (stringp (car x)) (equal (car x) y)) (implies (stringp (cdr x)) (stringp y)))
```

```
(implies (and (consp x)
              (stringp (car x))
              (equal (car x) y))
         (implies (stringp (cdr x))
                  (natp y)))
```

## A Few Words about Rev

- **Question 60** Define the concatenation function, `app`, so that `(app '(a b c) '(d e f))` is `(a b c d e f)`. •
- **Question 61** Define the reverse function, `rev`, so that `(rev '(a b c))` is `(c b a)`. •

If we can't write a 4-line program that works correctly, what hope is there?

And while `rev`, `rev1`, and `rev2` all seem to work, how do we know there is not some test case out there that shows they have bugs too?

But how do we know? Can we show they're the same?

We will learn to use logic to prove the mathematical theorem:

```
(and (equal (rev x) (rev1 x nil))  
      (equal (rev x) (rev2 x)))
```

## Next Week's Homework

On Question 101, you may assume  $x$  is a well-formed formula.

**Note:** You should not need the definition of  $wff$  to do any of the problems!