CSU290 Lecture Notes   Lecture 20   3 Nov 2008

Peter Dillinger

Given

```
(equal (app x y)
       (if (endp x)
         y
         (cons (car x)
               (app (cdr x) y))))

(equal (true-listp x)
       (if (endp x)
         (equal x nil)
         (true-listp (cdr x))))
```

Prove

```
(implies (and (endp x)
              (true-listp x))
         (equal (app x nil)
                x))
```

```
Assumptions: (endp x)
             (true-listp x)

(app x nil)
= { Def app }
(if (endp x)
    nil
    (cons (car x)
          (app (cdr x) y)))
= { Assumption (endp x) }
nil
= { Lemma 1, see below }
x
```

I need to prove a Lemma I claimed in that proof:

```
(implies (and (endp x)
              (true-listp x))
         (equal x nil))
```

I will do something a little different from the usual and start with the assumptions and work forward to the conclusion:

```
t
=> { Assumption }
(true-listp x)
=> { Def. true-listp }
(if (endp x)
  (equal x nil)
  (true-listp (cdr x)))
=> { Assumption (endp x) }
(equal x nil)
```

Having proved Lemma 1, and using that to prove (app x nil) = x under the assumptions, the proof is complete.

Now let’s prove

```
(implies (and (consp x)
              (true-listp x)
              (equal (app (cdr x) nil)
                     (cdr x)))
         (equal (app x nil)
                x))
```

```
Assumptions: (consp x)
             (true-listp x)
             (equal (app (cdr x) nil) (cdr x))
```

```
(app x nil)
= { Def app, Assumption (consp x) }
(cons (car x)
      (app (cdr x) nil))
= { Assumption (equal (app (cdr x) nil) (cdr x)) }
(cons (car x)
      (cdr x))
= { Cons axiom, Assumption (consp x) }
x
```

Here’s the new Cons axiom I just used:

```
(implies (consp x)
         (equal (cons (car x)
                      (cdr x))
                x))
```