27. Prove {app-assoc} theorem

(append [$x_1 \ldots x_n$] (append ys zs)) = (append (append [$x_1 \ldots x_n$] ys) zs)

(append [$x_1 \ldots x_n$] (append ys zs))

= (append (cons x_1 [$x_2 \ldots x_{n+1}$] (append ys zs)) {cons}

= (cons x_1 (append [$x_2 \ldots x_{n+1}$] (append ys zs)) {app1}

= (cons x_1 (append (append [$x_2 \ldots x_{n+1}$] ys) zs)) {inductive hyp}

= (append (cons x_1 (append [$x_2 \ldots x_{n+1}$] ys) zs)) {app1}

= (append (append (cons x_1 [$x_2 \ldots x_{n+1}$] ys) zs)) {app1}

= (append (append [$x_1 \ldots x_{n+1}$] ys) zs) {cons}

29. (defun rep (n x)

(if (zp n)

Nil

(cons x (rep (- n 1) x))))

Prove {rep-len}

(len (rep n x)) = n

30.

(member-equal y nil) = nil {mem0}

(member-equal y (cons x xs)) = (equal y x) OR (member-equal y xs)) {mem1}

Prove {member-equal}

(member-equal y (rep n x)) IMPLIES (member-equal y (cons x nil))

31. Prove {drop-all0}

(len (nthcdr (len xs) xs)) = 0 P(n)

Base case: n = 0

(len (nthcdr (len nil) nil)) = 0

(len (nthcdr (0) nil)) = 0 {len0}

(len (nil)) = 0 {sfx0}

0 = 0 {len0}

Inductive Hypothesis: For all n, P(n)

(len (nthcdr (len [x_1 … x_{n+1}]) [x_1 … x_{n+1}])) = 0

(len (nthcdr (len (cons x_1 [x_2 … x_{n+1}])) cons x_1 [x_2 … x_{n+1}])) {cons 2x}

(len (nthcdr (+ 1 (len [x_2 … x_{n+1}])) cons x_1 [x_2 … x_{n+1}])) {len1}

(len (nthcdr (+ (len [x_2 … x_{n+1}]) 1) cons x_1[x_2 … x_{n+1}])) {+ commutative}

(len (nthcdr (len [x_2 … x_{n+1}])) cons x_1 [x_2 … x_{n+1}])) {sfx1, rest}

(len (nthcdr (len [x_2 … x_{n+1}])) [x_2 … x_{n+1}])) { }

0 = 0 {Inductive Hypothesis}