PART A

1. Prove:

(append (append xs ys) zs) = (append xs (append ys zs))

PART A : BASE CASE { xs ⟶ '() }

Assume xs = '()

(append (append xs ys) zs)

= { by assumption, xs = '() }

(append (append '() ys) zs)

= { by append-nil law }

(append ys zs)

= { by append-nil law from right to left }

(append '() (append ys zs))

= { by assumption xs = '() }

(append xs (append ys zs))

PART B : INDUCTIVE HYPOTHESIS

Given xs = (cons a as)

Then inductive hypothesis:

(append (append as ys) zs) =

(append as (append ys zs))

PART C: INDUCTIVE CASE (xs $\mapsto$ (cons a as))

(append (append xs ys) zs)

= { by assumption xs = (cons a as) }

(append (append (cons a as) ys) zs)

= { by append - cons law }

(append (cons a (append as ys)) zs)

= { by append - cons law }

(cons a (append (append as ys) zs))

= { by inductive - hypothesis }

(cons a (append as (append ys zs)))

= { by append - cons law from right to left }

(append (cons a as) (append ys zs))

= { by assumption, xs = (cons a as) }

(append xs (append ys zs))

PART D: CONCLUSION

Thus from the above inductive proof, we can conclude that appending lists is an associative operation,

i.e

(append (append xs ys) zs) = (append xs (append ys zs))