structure List : LIST =

struct

val op + = InlineT.DfltInt.+

datatype list = datatype list

exception Empty = Empty

fun last [] = raise Empty

| last [x] = x

| last (\_::r) = last r

fun loop ([], []) = EQUAL

| loop ([], \_) = LESS

| loop (\_, []) = GREATER

| loop (x :: xs, y :: ys) =

(case compare (x, y) of

EQUAL => loop (xs, ys)

| unequal => unequal)

in

loop

end

end (\* structure List \*)