;Program 8

;Austin Spencer

;Dot production

;Helper functions

;Helper for part c

;Gets each individual product in a list

(define (product list1 list2)

(if (null? list1)

()

(cons (\* (car list1) (car list2))(product (cdr list1)(cdr list2)))

)

)

;Gets the smallest integer out of a given list

(define (minProd list)

(cond ((null? (cdr list)) (car list))

((< (car list) (minProd (cdr list))) (car list))

(#t (minProd (cdr list)))

)

)

;a) return the dot-production

(define (dotProduction list1 list2)

(if (null? list1)

0

(+ (\* (car list1) (car list2))

(dotProduction (cdr list1) (cdr list2))

)

)

)

;b) return whether or not the list contains duplicate entries

(define (isDuplicate list)

(if (null? list)

#f

(if (null? (cdr list))

#f

(if (= (car list) (car (cdr list)))

#t

(isDuplicate (cdr list))

)

)

)

)

;c) return the smallest product between corresponding pairs of numbers

;uses the two helper functions to return the smallest product

(define (smallProduct list1 list2)

(null? list1)

()

(minProd (product list1 list2))

)