2) (define (concatL l1 l2)(cond ((null? (car l1))'())

((null? (cdr l1))(list(string-append (car l1) (car l2))))

(else (cons (string-append (car l1) (car l2)) (concatL (cdr l1) (cdr l2))))))

3) (define (buildListh N E)

(cond ((= N 0) (list E))

(else (cons E (buildListh (- N 1) E)))))

(define (buildList N E)(display(buildListh N E)))

4) (define (listpicket P L)

(cond ((null? P) ((null? L)'()))

((null? (cdr L)) (cons (car L) (list P)))

(else (cons P (cons (car L)(listpicket P (cdr L)))))))