---------- PRB 1

Sequence recursive:
(sequence-rec 0)->1
(sequence-rec 1)->2
(sequence-rec 2)->5
(sequence-rec 3)->17
(sequence-rec 4)->107
(sequence-rec 8)->85691213438975

Sequence iterative:
(sequence-iter 0)->1
(sequence-iter 1)->2
(sequence-iter 2)->5
(sequence-iter 3)->17
(sequence-iter 4)->107
(sequence-iter 8)->85691213438975

---------- PRB 2

a) All function:
(all odd '(2 4 6))->#f
(all odd '(1 3 5 6)->#f
(all odd '()->#t
(all odd '(1 3 5 7)->#t

b) Exists function:
(exists odd '(2 4 5 6))->#t
(exists odd '(2 4 6))->#f
(exists odd '())->#f
(exists odd '(1))->#t

(for odd = a function that tests if a number is odd or not)

---------- PRB 3

FIXED_LENGTH = 3: (fixed-length-even-list '((1 2 3) (2 2 2)))->#t
FIXED_LENGTH = 3: (fixed-length-even-list '((2 4 2) (1 2 3)))->#t
FIXED_LENGTH = 3: (fixed-length-even-list '((2 2 2)))->#t
FIXED_LENGTH = 3: (fixed-length-even-list '((2 2 3) (3 4 5)))->#f
FIXED_LENGTH = 2: (fixed-length-even-list '((1 2) (2 2)))->#t
FIXED_LENGTH = 2: (fixed-length-even-list '())->#f
FIXED_LENGTH = 2: (fixed-length-even-list '((2 3) (2 2 4)))->#f

---------- PRB 4

for f(x)=2*x : ((f-to-n f 0) 2)->2
for f(x)=2*x : ((f-to-n f 1) 2)->4
for f(x)=2*x : ((f-to-n f 2) 2)->8
for f(x)=2*x : ((f-to-n f 3) 2)->16
for f(x)=2*x : ((f-to-n f 4) 2)->32

---------- PRB 5

for f(x)=2*x : (integral f 0)->0
for f(x)=2*x : (integral f 1)->1
for f(x)=2*x : (integral f 2)->4
for f(x)=2*x : (integral f 3)->9
for f(x)=2*x : (integral f 5)->25