CS 270 - Lab 8

M. Boady, B. Char, J. Johnson, G. Long, S. Earth

1 Introduction

You may work in teams of **one** or **two** students. Submit one copy for the entire group.

Write your answers on this lab sheet. Only what is written on this lab sheet will be graded.

This lab is due at the end of the class period. You may not continue to work on it once class has ended.

This lab contains 4 questions.

Grading

- 25 points - Putting everyones names on this page
- 20 points - Earned for each correct question (Answer is fully correct)
- 5 points - Earned for **partial credit** any question

No additional point amounts can be earned. You cannot earn 7 points on a question for example.

The maximum score for a lab is 100. If you get everything correct, that adds up to 105 points but will be reduced to 100.

A question will be marked correct as long as it covers all requirements of the question. It does not need to be perfect, but must be fully correct. A single typo or very minor issue where the intention is clear and all requirements are met would still earn full points.

We want you to complete questions fully, not try to earn partial credit on multiple questions. You may ask your Professor/Course assistant questions during lab.

Enter the name of the student in the group

Member 1: Elan Rubin

Member 2: I usually work with Josh Koo, but for this lab we had problems with ProofBuddy and were told to do the lab separately as homework.

Score (out of 100): _____ **Graded By**: _____

Question 1 :

The following function definition is used in this question.
Input Contract: a and b are nonnegative integers
Output Contract: (F a b) is a recursively defined integer **pattern redacted for students**

```
(define (F a b)
   (if (= b 0)
       a
       (+ 1 (F a (- b 1))))
   )
)
```

Use **Equational Reasoning** to show how (F 5 1) evaluates.

Note: Minor mistakes like extra parenthesis will be ignored when grading.

```
;(Question 1)
(define (F a b)
  (if (= b 0)
      a
      (+ 1 (F a (- b 1)))))

;(F 5 1) premise
;(if (= 1 0) 5 (+ 1 (F 5 (- 1 1)))) expand definition
;(if #f 5 (+ 1 (F 5 (- 1 1)))) evaluate condition
;(+ 1 (F 5 (- 1 1))) take false branch
;(+ 1 (F 5 0)) evaluate subtraction
;(+ 1 (if (= 0 0) 5 (+ 1 (F 5 (- 0 1))))) expand f
;(+ 1 (if #t 5 (+ 1 (F 5 (- 0 1))))) evaluate condition
;(+ 1 5) take true branch
;6 final result
```

Question 2 :

The following function definition is used in this question.
Input Contract: a and b are nonnegative integers
Output Contract: (H a b) is a recursively defined integer **pattern redacted for students**

```
(define (H a b)
  (if (< a b)
      0
      (+ 1 (H (- a b) b))
  )
)
```

Use **Equational Reasoning** to show how (H 3 2) evaluates.

Note: Minor mistakes like extra parenthesis will be ignored when grading.

```
;(Question 2)
(define (H a b)
  (if (< a b)
      0
      (+ 1 (H (- a b) b))))

;(H 3 2) premise
;(if (< 3 2) 0 (+ 1 (H (- 3 2) 2))) expand definition
;(if #f 0 (+ 1 (H (- 3 2) 2))) evaluate condition
;(+ 1 (H (- 3 2) 2)) take false branch
;(+ 1 (H 1 2)) evaluate subtraction
;(+ 1 (if (< 1 2) 0 (+ 1 (H (- 1 2) 2)))) expand h
;(+ 1 (if #t 0 (+ 1 (H (- 1 2) 2)))) evaluate condition
;(+ 1 0) take true branch
;1 final result
```

Question 3 :

The following function definition is used in this question.
Input Contract: a and b are nonnegative integers
Output Contract: (Q a b) is a recursively defined integer **pattern redacted for students**

```
(define (Q a b)
  (if (= b 1)
      a
      (+ a (Q a (- b 1))))
  )
)
```

Use **Equational Reasoning** to show how (Q 3 2) evaluates.

Note: Minor mistakes like extra parenthesis will be ignored when grading.

Special: This lab has an extra credit on the next page!

```
;(Q 3 2) premise
;(if (= 2 1) 3 (+ 3 (Q 3 (- 2 1)))) expand definition
;(if #f 3 (+ 3 (Q 3 (- 2 1)))) evaluate condition
;(+ 3 (Q 3 (- 2 1))) take false branch
;(+ 3 (Q 3 1)) evaluate subtraction
;(+ 3 (if (= 1 1) 3 (+ 3 (Q 3 (- 1 1))))) expand q
;(+ 3 (if #t 3 (+ 3 (Q 3 (- 1 1))))) evaluate condition
;(+ 3 3) take true branch
;6 final result

;(Question 4)
(define (M x L)
  (if (= x 0)
      (first L)
      (M (- x 1) (rest L))))
```

Question 4 :

The following function definition is used in this question.
Input Contract: x is a nonnegative integer, and L is a list of positive integers
Output Contract: (M x L) is a recursively defined integer **pattern redacted for students**

```
(define (M x L)
  (if (= x 0)
      (first L)
      (M (- x 1) (rest L))))
```

Use **Equational Reasoning** to show how (M 1 '(1 2 3)) evaluates.

Note: Minor mistakes like extra parenthesis will be ignored when grading.

```
51 ;(Question 4)
52 (define (M x L)
53   (if (= x 0)
54       (first L)
55       (M (- x 1) (rest L))))
56
57 ;(M 1 '(1 2 3)) premise
58 ;(if (= 1 0) (first '(1 2 3)) (M (- 1 1) (rest '(1 2 3)))) expand definition
59 ;(if #f (first '(1 2 3)) (M (- 1 1) (rest '(1 2 3)))) evaluate condition
60 ;(M (- 1 1) (rest '(1 2 3))) take false branch
61 ;(M 0 '(2 3)) evaluate subtraction and rest
62 ;(if (= 0 0) (first '(2 3)) (M (- 0 1) (rest '(2 3)))) expand m
63 ;(if #t (first '(2 3)) (M (- 0 1) (rest '(2 3)))) evaluate condition
64 ;(first '(2 3)) take true branch
65 ;2 final result
```

Question 5 :

This question only gives extra credit. You cannot lose points for attempting it.

You will earn 1 point extra credit for each correct answer.

Each function in this lab was *inspired* by a math function. Functions like subtract, addition, remainder, quotient, multiplication, exponent, log, square root, etc.

Can you determine what each function was based on?

(a) (1 point) What math function is F inspired by?

The function F looks like it's addition.

(b) (1 point) What math function is H inspired by?

The function H looks like integer division.

(c) (1 point) What math function is Q inspired by?

The function Q looks like multiplication.

(d) (1 point) Describe what the function M is doing.

I'm not sure, but function M might be doing list indexing.