CS270: LAB #1

Racket Intro

You may work in groups of three people (two or four is acceptable in the event of an unscheduled absence). Ideally, your lab is submitted (properly tagged) into Gradescope by the end of class; however, the formal deadline is Friday at 11:59pm.
In order to receive credit, follow these instructions:

[a] Every team member should be discussing simultaneously the same problem – do NOT try to divvy up the labor and assign different problems to different students since the material is cumulative.

[b] Directly edit this lab PDF using Sedja with your answers (extra pages can be added in the rare event you need more than the allotted space)

[c] Each lab, rotate which member has the responsibility of being the Scribe. This is the person that is typing the answers and uploading the final PDF – note that only a single copy of the filled in PDF is turned into Gradescope. Only one lab needs to be submitted for the entire team, and all members receive the same score. Make sure to use a font that your PDF editor is compatible with (otherwise you might find your answers appear as weird shapes/sizes or simply disappear entirely!)

[d] The Gradescope submission must have each answer properly tagged with the appropriate question. Moreover, every member of the team must be listed as a submitter. Although it is the Scribe which executes these actions, it is still the responsibility of the entire team to make certain this is done properly (thus it is highly recommended that the Scribe share their screen so the entire team can witness it). Answers which are improperly tagged cannot be seen by the grader and thus cannot be scored.

[e] **FOR REMOTE ONLY**: Each lab, rotate which member has the responsibility of being the Recorder. This is the person who hits the Zoom Record button (once the technical permission is granted by the TA/RCF/Professor) and ensures that everyone has their camera/microphone on. They are also the member that is responsible to make sure the DrexelStream video is marked as viewable and entered into the https://tinyurl.com/VidLinkForm webform before 11:59pm (they should also email the rest of their team as confirmation.) Note that the video file doesn't get created/processed until after the Recorder has quit Zoom.

[f] Each lab, rotate which member has the responsibility of being the Manager. This is the person that ensures that everyone is participating equally and honestly, keeps the group on task, ensures that all team members understand a solution before going on to the next question, and presses the "hand up" button in Zoom to summon a TA or the professor (but they only do so after surveying the group to make sure everyone has the same question).

Team Name (CS pioneer): ____________________________

Scribe name: ____________________________

Recorder name: ____________________________

Manager name: ____________________________

Other team member (if any): ____________________________

Question 1: 5 pts

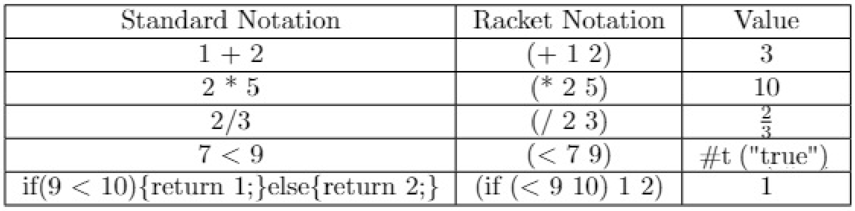
Read over the "Advice from last quarter's students" located in the Course Information tab from the lefthand nav panel.

List here at least three common threads among the advice offered.

Does anything in particular resonate with you based on experiences you have had in other courses?

Question 2: 10 points

The Racket Programming Language is a purely functional language. It has a syntax that is different from most programming languages. Each command starts and ends with parenthesis. The function name is always the first item inside the parenthesis. The function name is followed by any input arguments.



Standard Notation	Racket Notation	Value
1 + 2	(+ 1 2)	3
2 * 5	(* 2 5)	10
2/3	(/ 2 3)	$\frac{2}{3}$
7 < 9	(< 7 9)	#t ("true")
if(9 < 10){return 1;}else{return 2;}	(if (< 9 10) 1 2)	1

Write each of the below expressions in Racket Notation. Translate the expressions exactly, do not evaluate them.

(a) (2 points) 7 + 9 * 3

(b) (2 points) 90 < 100

(c) (2 points) (50 * 3) > 100

(d) (2 points) if (2 - 5 > 0) then return 12; else return 15;

(e) (2 points) 9 * 3 + 2 * 4

Question 3: 15 points [3 pts each]

What does each of the following expressions evaluate to? You can verify in DrRacket, but try it manually first.

(a) (+ (* 9 (+ 1 1)) 2)

(b) (- 8 9)

(c) (* 9 (/ 7 9))

(d) (if (< 1 2) 3 4)

(e) (if (> 8 9) 5 6)

3 Function Definitions

Racket is designed for recursive functions. Any iteration must be completed recursively. The factorial function is the product of all numbers between 1 and n.

$$n! = 1 * 2 * 3 * 4 \cdots * (n-1) * n \qquad = \begin{cases} 1 & n = 0 \\ n * (n-1)! & n > 0 \end{cases} \tag{1}$$

The Racket definition is provided below.

```
(define (fact n)
  (if (<= n 1) 1 (* n (fact (- n1)))))
```

The first few return values are

(fact 0)	1
(fact 1)	1
(fact 2)	2
(fact 3)	6

Question 4: 24 points
Answer each of the following questions. These questions help you walk through the implementation of fact to show why it works

(a) (3 points) What is value is returned by (fact (- 4 1))?

(b) (3 points) What value is return by (* 4 (fact (- 4 1)))?

(c) (3 points) Is (<= 4 1) true or false?

(d) (3 points) What is the return value of (fact 4)?

(e) (3 points) What is the return value of (fact 5)?

(f) (3 points) What is the return value of (fact 6)?

(g) (3 points) How many times does (<= n 1) return false when (fact 6) is executed?

(h) (3 points) How many times does (<= n 1) return true when (fact 6) is executed?

4 Recursive Thinking

Think about exponents. You may recall that a^0 =1 provided a≠0

The following questions should lead you to discover a recursive formula for exponents.

Do not answer in Racket syntax until the last partF.

Question 5: 21 points

(a) (3 points) What is the numerical value of 2^1?

(b) (3 points) Define 2^1 algebraically in terms of 2 and 2^0

(c) (3 points) What is the numerical value of 2^6?

(d) (3 points) Write 2^6 algebraically in terms of 2 and 2^5

(e) (3 points) In general, how would you write 2^n in terms of 2 and a previous power of 2?

(f) (6 points) Write a recursive Racket function (pow a b) which computes a^b and obeys the contracts specified below

```
; input contract = a is a positive integer, b is a non-negative integer
; output contract = (pow a b) is a^b
; example: (pow 2 3) would evalute to 8, since 2^3 = 2*2*2 = 8
```

5 Recursive Summation

Define a Racket function (S n) that sums up all numbers from 0 to n recursively. The below table shows the first few values for the function.

$S(0)$	0
$S(1)$	1
$S(2)$	3
$S(3)$	6
$S(4)$	10
$S(5)$	15

Question 6 : 16 points

Provide an implementation in Racket for the function (S n). Be sure to include comments for the input and output contracts

Question7: 9points

Set Up Discord:

1. Have Each Group Member Complete the below steps individually.
   (a) Join the CCI Discord server using directions listed on the Syllabus
   (b) Select the channel #introduction under CS270-SU23
   (c) Post a message telling the class: Your name, major, hometown, preferred pronoun, and something interesting about yourself (e.g. your favorite hobby, music, sport, movie, game, book, tv series, etc)
2. Submit a screenshot here of each of the messages