Quiz 3 functional programming Skills: Use of functional paradigm.

Name _____ **Student Number**_____

Problem:

For whatever reasons you might think, you have been asked to help build a Data Structure API in scheme. Implement one or more of the following data structures, for each data structure implement its basic operations such as add, remove, search, etc... If you think it necessary, you can add more functions as long as they are related to the data structure you choose.

matrix (addition of matrices, subtraction of matrices, dot product)

stack (push pop peek)

queue (enqueue dequeue front)

graph (add arc, add node, search)

heap (add remove heapify)

linked list (add first, add last, add before, search)

Instructions:

Use the problem above to show what you know. Make a **small diagram of your solution or write the general algorithm in pseudo code.** Then implement it in racket/scheme. Submit your code and the diagram to canvas before the end of the class.

Include the examples of calls stated above to show your code works as well as comments in the code. If I can't run code or I don't understand how your functions should be called, they will be deemed as wrong.

Using scheme built-in functions will provide NO evidence (e.g. map, member, append, reverse, etc...), however, **you can reimplement them if you wish.**

If your algorithms are ambiguous or they do not match your code, you will get 0 in the quiz and a possible F.I.A.

You can only use your notes, you cannot check any online **resources, previous labs, electronic devices or look at other peoples shared screens**. Any use of external devices be considered F.I.A.

Your grade will depend on the competence you demonstrate in your solution for the following element, your program must works correctly even if it is a partial solution:

	No Evidence 0	Basic 10	Advanced 30
Use of lambda Function	____	____	____
Use of Recursion	____	____	____
Processing of List	____	____	____
Mutual Recursion	____	____	____
Functional Purity (Modelling in paper)	____	____	____