Midterm (50 pts)

Write the answers only in the space provided. Use the “blank” side(s) of each page for scratch work, especially to show evaluation details where appropriate. Be concise, precise, and legible.

<table>
<thead>
<tr>
<th>Name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>UID</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td></td>
</tr>
</tbody>
</table>
1 General (4 + 5 + 6 pts)

1. Explain clearly why FORTRAN designers decided not to allow recursion and dynamic storage allocation in the language?

2. Can letrec be simulated using let? If yes, provide a generic translation. If no, explain why not.
3. What is a strongly typed language? What is dynamic typing?
2 Recursive Definition and Evaluation (6 + 4 + 5 pts)

(define (whatsUp x)
  (cond ((null? x) 4)
        ((string? (car x)) 5)
        ((symbol? (car x)) (+ (whatsUp (cdr x)) 2))
        ((number? (car x)) (* (whatsUp (cdr x)) 10))
        (else (+ (whatsUp (car x)) (whatsUp (cdr x)))))
)

1. What is the value of (whatsUp '(c s 3 (7)))?

2. What is the value of (whatsUp '(r2d 2 "c3pu") )?
3. Informally, but precisely and concisely, describe the output of the above definition on a flat (that is, not nested) list of strings and numbers. (Hint: Use examples to improve your understanding.)
3 Trace Expression Evaluation in Detail (5 pts)

(apply append (append '(( 1"2")) (list (list 'cons (procedure? append) ))))
A set is an unordered collection of values of the same type, with no significance attached to duplicates. The operations on the ADT ISet, the integer sets, are informally described below. (Assume Int is type name for integers and Bool is type name for booleans.)

- **empty**: Yields the empty set.
- **insert**: Takes an integer and a set as inputs, and yields the set resulting from introducing the integer into the set.
- **isEmpty**: Takes a set as input, and checks if it is empty.
- **isMember**: Takes an integer and a set as inputs, and checks if the integer belongs to the set.
- **union**: Takes two sets as inputs, and yields the set containing integers from both sets. (That is, \( \text{union} \{1,2\}, \{2,3\} = \{1, 2, 3\} \).)
- **intersection**: Takes two sets as input, and yields the set containing integers common to both sets. (That is, \( \text{intersection} \{1,2\}, \{2,3\} = \{2\} \).)
- **size**: Takes a set as input, and yields the number of distinct integer members. (That is, \( \text{size} \{22,33\} = 2 \).)

1. Specify the signatures and classify the aforementioned operations on ADT ISet.
2. Write the axioms that \textit{size} operation must satisfy.

3. Write the axioms that \textit{intersection} operation must satisfy.

4. Assuming that an \texttt{ISet} instance is represented using a Scheme list of integers, write a Scheme function (implementation) for carrying out the \textit{union} operation. That is, assume \texttt{empty} is () and \texttt{insert} is \texttt{cons}. 