1. Suppose the Global Environment contains the following entries (besides all the built-in functions and variables it normally contains).

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>C</td>
<td>6</td>
</tr>
<tr>
<td>X</td>
<td>9</td>
</tr>
<tr>
<td>fun</td>
<td>(lambda (X Y) (* X (+ Y 4)))</td>
</tr>
</tbody>
</table>

Using the Global Environment entries in the table above, show the result of evaluating each of the following expressions. Keep in mind that the evaluation may cause an error message (the > is the Interactions Window prompt). If you think the expression causes an error, write ERROR and explain why you think the expression causes an error. Otherwise, write the value returned.

> (* B 1000)
2000

> (fun B)
**ERROR:** fun has 2 input parameters

> (string? X)
#f

> (fun A X)
= (* A (+ X 4)) = (* 3 (+ 9 4)) = 39

> (string=? X "6")
**ERROR:** string=? takes only string arguments

*Problem 2 on the back of this sheet.*
2. Consider the following 2 example lists: EVENS and THREES:

;; Examples
(define EVENS (cons 2 (cons 4 (cons 6 (cons 8 (cons 10 empty))))))
(define THREES (cons 3 (cons 6 (cons 9 (cons 12 (cons 15 empty))))))

Show the result of evaluating the following expressions, keeping in mind that the evaluation may result in an error. If you think the statement will cause an error, just write ERROR for the result of evaluation.

(first (rest THREES)) ==> 6

(rest (rest (rest (rest EVENS)))) ==> (10)