NAME: 

Instructions:

1) This test is 6 pages in length.

2) You have 75 minutes to complete and turn in this test.

3) Short-answer questions include a guideline for how many sentences to write. Respond in complete English sentences.

4) This test is closed books, notes, papers, friends, neighbors, etc.

5) Use the backs of pages in this test packet for scratch work. If you write more than a final answer in the area next to a question, circle your final answer.

6) Write and sign the following: “I pledge my Honor that I have not cheated, and will not cheat, on this test.”

________________________________________________________________________

________________________________________________________________________

Signed: ______________________________________________
1. [5 points]
What does it mean for a PL to be weakly typed? [1-2 sentences]

2. [75 points]
Let L be a call-by-name, simply typed lambda calculus with base type int. The syntax for expressions in L is:  
\[ e::= e \, e' \mid x \mid n \mid \lambda x:\tau. e \]

(a) Using the following SML definitions for L:
\begin{verbatim}
datatype typ = Int | Arr of typ*typ;
datatype exp = App of exp*exp | Var of string | Int of int |
| Lam of string*typ*exp;
exception captured;
\end{verbatim}

implement an SML function \( \text{sub}: \text{exp} \to \text{string} \to \text{exp} \to \text{exp} \) such that \( \text{sub} \ e \ x \ e' \) returns \([e/x]e'\) unless a variable gets captured, in which case \textit{captured} gets raised.
(b) Define L’s static semantics.

(c) Define L’s SOS-style dynamic semantics $\langle e \rightarrow e' \rangle$.

(d) Define L’s big-step operational semantics $\langle e \downarrow v \rangle$.

(e) Formally state all the standard type-safety lemmas/theorems/corollaries for L.
(f) Using your rules for L, as well as the following:

\[
\begin{array}{c}
\text{Re} \\
\text{Tr}
\end{array}
\]

e 
\rightarrow* e' 

prove that \( \forall e, v: (e \rightarrow* v) \Rightarrow (e \downarrow v) \).
3. [20 points]

a) Encode a ternary logic (three-valued logic) into the untyped lambda calculus. The three values are: T (true), F (false), and B (both). Besides for the values themselves, provide encodings for AND(v1,v2) and OR(v1,v2), defined below. Both operations must be short circuit (lazy).

<table>
<thead>
<tr>
<th>v1</th>
<th>v2</th>
<th>AND(v1,v2)</th>
<th>OR(v1,v2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>B</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>B</td>
<td>B</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>B</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>B</td>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>T</td>
<td>B</td>
<td>T</td>
</tr>
<tr>
<td>B</td>
<td>F</td>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>B</td>
<td>B</td>
<td>B</td>
<td>B</td>
</tr>
</tbody>
</table>

b) Using the call-by-value strategy and your response to Part (a), trace the evaluation of AND(B,OR(F,T)) to B. Show each step of the evaluation and underline redexes.
4. [12 points]
Returning to the language L of Problem 2, assume the following lemmas hold.

Lemma 1. \( \forall e_1, e_2, e_3: (e_1 \rightarrow \ast e_2 \land e_2 \rightarrow \ast e_3) \Rightarrow (e_1 \rightarrow \ast e_3) \)

Lemma 2. \( \forall e_1, e_1', e_2: (e_1 \rightarrow \ast e_1') \Rightarrow (e_1 e_2 \rightarrow \ast e_1' e_2) \)

Now prove that \( \forall e, v: (e \uparrow v) \Rightarrow (e \rightarrow \ast v) \).