CMSC330 Fall 2009 Quiz #2

Name ________________________________

Discussion Time (circle one): 10am 11am 12pm 1pm 2pm 3pm

Instructions

- You have 20 minutes for this quiz.
- This is a closed book exam. No notes or other aids are allowed.
- For partial credit, show all of your work and clearly indicate your answers.
- Write neatly. Credit cannot be given for illegible answers.

1. (6 pts) OCaml Types and Type Inference
   a. (2 pts) Give the type of the following OCaml expression
      
      \[
      \text{fun x -> [ x+2 ]} \quad \text{Type =}
      \]

   b. (2 pts) Write an OCaml expression with the following type
      
      \[
      \text{int -> bool} \quad \text{Code =}
      \]

   c. (2 pts) Give the value of the following OCaml expression. If an error exists, describe the error.
      
      \[
      \text{let x = (fun z -> z – 1) in x 1} \quad \text{Value =}
      \]

2. (6 pts) OCaml Programming

   Using fold and an anonymous function, write a function \textit{numAdults} which when applied to a list of ints \textit{lst}, returns the number of elements of \textit{lst} that are 18 or over. Example:

   \[
   \text{numAdults [17;18; 21;16; 25] = 3}
   \]

   \[
   \text{let rec fold f a l = match l with}
   \]
   \[
   [] -> a
   \]
   \[
   | (h::t) -> fold f (f h) t
   \]
3. (5 pts) First Sets

Compute First sets for S and A in the following grammar:

\[ S \rightarrow Ab \quad A \rightarrow dA \]
\[ S \rightarrow c \quad A \rightarrow \varepsilon \quad (*) \text{epsilon} (*) \]

4. (3 pts) Parsing

Finish writing a predictive, recursive descent parser for the following grammar.

\[ S \rightarrow aSb \]
\[ S \rightarrow \varepsilon \quad (*) \text{epsilon} (*) \]

You may use the following utilities

<table>
<thead>
<tr>
<th>lookahead</th>
<th>Variable holding next terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td>match (x)</td>
<td>Function to match next terminal to x</td>
</tr>
<tr>
<td>error ()</td>
<td>Reports parse error for input</td>
</tr>
</tbody>
</table>

```c
parse_S() {
    if (lookahead == "a") {
        // your code here
    } else {
        // your code here
    }
}
```