CMSC330 Fall 2011 Example Quiz #4 Solution

1. (14 pts) Parsing

   Consider the following grammar: S → aA | A A → bS | ca

   a. (6 pts) Compute First sets for S and A

   First(A) = First(bS) ∪ First(ca) = {b, c}

   First(S) = First(aA) ∪ First(A) = {a, b, c}

   b. (12 pts) Write a predictive, recursive descent parser for the grammar

```
parse_S( ) {
        if (lookahead == "a") {        // S → aA
                match("a");
                parse_A( );
        }
        else if ((lookahead == "b") ||
                 (lookahead == "c")) { // S → A
                parse_A( );
        }
        else error( );
}
parse_A( ) {
        if (lookahead == "b") {        // A → bS
                match("b");
                parse_S( );
        }
        else if (lookahead == "c") {  // S → aA
                match("c");
                match("a");
        }
        else error( );
}
```

2. (16 pts) Lambda calculus

Circle all free (unbound) variables in the following λ-expression

a. (2 pts) (λa. a (λb. b a) a b) a

(λa. a (λb. b a) a b) a

Evaluate the following λ-expressions as much as possible. Show each beta-reduction

b. (3 pts) (λx.λy.y x) a (λz.z) b

(λx.λy.y x) a (λz.z) b → (λy.y a) (λz.z) b → ((λz.z) a) b → a b

c. (3 pts) (λy. λx. x y) x a b

(λy. λx. x y) x a b → (λy. λz. z y) x a b → (λz. z x) a b → a x b

d. (8 pts) Using encodings, show 3*1 =>* 3. Show each beta-reduction.

=>* indicates 0 or more steps of beta-reduction

M * N = λx.(M (N x))
1 = λf.λy.f y
2 = λf.λy.f (f y)
3 = λf.λy.f (f (f y))
4 = λf.λy.f (f (f (f y)))

3*1 => **λx.(3 (1 x))**
=> λx.(3 ((λf.λy.f y) x))
=> λx.(3 (λy.x y))
=> λx.(λf.λy.f (f (f y)) (λy.x y))
=> λx.(λy.(λy.x y) ((λy.x y) ((λy.x y) y)))
=> λx.(λy.(λy.x y) ((λy.x y) (x y)))
=> λx.(λy.(λy.x y) (x (x y)))
=> λx.(λy.x (x (x y)))
=> 3