Programming Language Concepts, CS2104
Tutorial 4 (16 September 2011)
(All students must prepare/attempt in advance.)

Exercise 1

Build a syntax analyzer (without building an AST) for the language generated by the following grammar:

```
<S> ::=  ‘(‘ <A> ‘)’
<A> ::=  ‘[‘ <A> ‘]’
       | ‘{‘ <A> ‘}’ <S>
       | a | … | z
```

Exercise 2

Explain the following C declaration:

```
int **(*(*p)())[10] ;
```

Assume also the following declaration:

```
int **(*f())[] {

    /* to be filled in */

}
```

Is the assignment

```
p = &f
```

legal? Fill in the body of `f` with code that would be accepted by the compiler.
.

Exercise 3

Add a factorial operator to the language of expressions defined in Lecture 5. Its precedence level should be higher than that of all other operators. The operator should also be associative, meaning that `a!!` should be interpreted as `(a!)!`. Write the grammar rules and their formulation as inference rules for the new language.

Exercise 4

Modify the syntax analyzer discussed in Lecture 5 so that it can accept the language you defined in Exercise 3, and generate an AST for correct expressions.