

COSC 480 – Compilers
Spring 2009
Final Exam Review

- 1.) Define what an LL(1) grammar is. What are the 3 conditions that must hold for a grammar to be LL?
- 2.) Define what an LR(1) grammar is. What kind of parsing is typically used for an LR grammar?
- 3.) Define FIRST and FOLLOW and the algorithms to determine each of those sets.
- 4.) Describe the algorithm for determining the predictive parsing table for an LL grammar.
- 5.) Give the 6 steps involved in a modern compiler and describe each of the steps.
- 6.) List and describe the five classes of tokens that cover almost all tokens for modern programming languages.
- 7.) Divide the following statement:

$$E = M * C ** 2$$

Into appropriate token pairs following the patterns in figure 3.2 (pg 112) in your book. (On the exam I would provide you the table for the patterns).

- 8.) List three possible error-recovery actions for a lexical error.
- 9.) Describe the languages denoted by the following regular expressions:

$$\begin{aligned} &a(a|b)^*a \\ &a^*ba^*ba^*ba^* \\ &(a|b)^*a(a|b)(a|b) \end{aligned}$$

10.) Develop the regular expression for the language containing all strings that start with an a and has an even length or starts with a b and has an odd length.

- 11.) Create the 3 address code for the following arithmetic expression:

$$a + a * b - c / d * (a + b)$$

- 12.) Give the quads for the following expression:

$$a = a + b / c$$

13.) Create the 3 address code for the following snippet with short-circuit evaluation:

```
if (a < (b + c) && a != b)
    goto b
else
    a = a + b
```

14.) Give the quads for the following snippet:

```
switch (a){
    case 1:
        a = a + b;
        break;
    case 2:
        b = b + c;
        break;
    default:
        c = c + a;
        break;
}
```

15.) Define peephole optimization and the four optimizations it can be used for.