

CSL 862 / CS 902 Course Assignment 1.

Regular Expressions and DFA

Draw DFAs for the following languages (5 + 5 + 5)

1. $L = \{w \in \{0, 1, 2, 3, 4\}^* \mid w \text{ has no repeated digits}\}$
2. $L = \{w \in \{1, 2, 3\}^* \mid \text{Number of 2's modulo 2 equal the number of 3's modulo 3}\}$
3. $L = \{w \in \{fopen, fclose, fread, fwrite\}^* \mid w \text{ denotes a valid set of file operation}\}$.

Bonus: Write the equivalent REs. (10)

CFG

Write the CFG for the following language: (5 + 5 + 5)

1. $L = \{w \in \{0, 1\}^* \mid w \text{ contains equal number of 0s and 1s}\}$.
2. $L = \{w \in \{0, 1\}^* \mid w \text{ contains unequal number of 0s and 1s}\}$.
3. $L = \{w \in \{push, pop, top\}^* \mid w \text{ denotes a valid set of stack operations}\}$.

Parsing

LL(1) Grammar (30), Parser Implementation (40).

Consider the grammar

$\text{stmt} ::= \text{id}() \mid \text{stmt} ; \text{stmt} \mid \{ \text{stmt} \} \mid \text{if} (\text{id}) \text{stmt} \text{ else } \text{stmt}$

where stmt is the only non-terminal symbol, stmt is the start symbol, and

$\text{id} () ; \{ \}$ if else

is the list of terminal symbols. The terminal symbol id is defined using the regular expression (letter^+) where letter is an ascii character in the interval $a \cdot \cdot \cdot z$. The grammar generates a subset of the Java statements.

Rewrite the grammar into a grammar which is LL(1), and use the rewritten grammar as the basis for implementing a recursive descent parser in Java.

Write the LL(1) grammar, the FIRST and FOLLOW sets for each non-terminal symbol, and the predictive parsing table, together with an argument that the grammar is LL(1).

Submit electronically your program. Your main file should be called Parse.java, and if Statement is a file, possibly containing a Java statement according to the above grammar, then

```
java Parse < Statement
```

outputs
either "Statement parsed successfully",
or "Parse error".