CS 241 Tutorial 5

CS 241 Tutor

February 10, 2012

1 Regular Expression (RE)

- ϵ
  The empty word. NOTE: this is different from $\emptyset$.
- R_1R_2 where $R_1 \in RE$ and $R_2 \in RE$
  words matched by R_1 followed by words matched by R_2
- $R_1|R_2$ where $R_1 \in RE$ and $R_2 \in RE$
  words matched by R_1 OR by R_2. Represents Choice
- R^* where $R \in RE$
  0 or more words each of which are matched by R

Theorem:
For a language L the following are equivalent:

- $\exists$ DFA accepting L
- $\exists$ RE accepting L
- $\exists\ \epsilon - NFA$ accepting L
- $\exists$ NFA accepting L

What is the significance of this theorem?

Exercises

1. Write the regular expression that recognizes only the empty word.
2. Write a regular expression that recognizes a.
3. If $L = \emptyset$, is ε a word in the language?
4. Is 10000010 a word specifed by regular expression $(0^*10)^*$?
5. Write a regular expression that recognizes $\{xx, xy, yx, yy\}$.
6. Is ε specified by this regular expression: $(a^*b)^*ba^*$.
7. Let $\Sigma = \{G, C, A, T\}$. Let $L =$*all strings containing GACAT*. Construct a regular expression that specifies this lanaguage.

8. Let $\Sigma = \{a, b, c\}$. Let $L = \left\{x \in \Sigma^* | \exists i, j, k : x = a^i b^{2j} c^k\right\}$. Construct a regular expression that specifies this language.

9. Let $\Sigma = \{0, 1\}$. Let $L = \{\text{strings over } \Sigma \text{ of odd length}\}$. Construct a regular expression that specify this language.

10. Let $\Sigma = \{0, 1\}$. Let $L = \{x \in \Sigma^* : x \text{ is of length at least 4 and starts OR ends with 11, but not both}\}$. Construct a regular expression that will recognize this language.

11. $\Sigma = \{0, 1\}$. Write a regular expression that will accept binary strings that contain at least two occurrences of '00'.

12. $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, .\}$ Write a regular expression that will accept floating point numbers. That is, all numbers that are positive, negative, or ze ro, and have a point '.' separating its whole and fractional parts. (eg. -1.0, 0.0, 123.2352, -123.0000001, -00005.0, ...)

2 Scanning

Maximal Munch is a scanning algorithm that consumes as much of the input as possible to tokenize a string of text. Maximal Munch makes certain restrictions on the structure of the underlying DFA. What are these restrictions? Why are these restrictions neccessary?

Using the given Moore machine, scan the following strings using the Maximal Munch algorithm and note all of the tokens returned.

- "1637cherry7382–"
- "0x012345789-0x123-12345
- "please endthis"