Regexs with DFA

CS230 Tutorial 11
Regular Expressions (Regex)

This way of representing regular languages using metacharacters. Here are some of the most important ones to know:

- **|** -- OR
  - example: a | b -- this represents a OR b

- **()** -- grouping, basically like regular parentheses in math
  - example: m(i | a)lk -- this represents ‘milk’ OR ‘malk’

- **[]** -- matches one character in the list specified in the brackets
  - example: [abc] -- this represents a OR b OR c
  - can use - to specify a range in the brackets, i.e. [a-c] is the same as [abc]

- **^** -- NOT

- **.** -- any ONE character
  - example: a.c -- this represents a[every ascii character]c
Regex continued

More metacharacters...

- ? -- the previous character 0 or 1 time
  - example: aa? -- this means a OR aa
- + -- the previous character 1 or more times
  - example: a+ -- this matches a OR aa OR aaa OR aaaa OR ….. infinite a’s
- * -- the previous character 0 or more times
  - example: a* -- this matches ε OR a OR aa OR aaa OR ….. infinite a’s
- \ -- escape: you use this when you want to treat one of the metacharacters as a normal ASCII character
  - example: a\* -- this means a* literally, not aa or aaa or …
  - example: \ -- this means the backslash character
Regex Examples

Make a regex for the language of any combination of the letters a and b.

Make a regex for the language of any number of a’s, followed by an even number of c’s.

Describe the language (with an english sentence) represented by: a+[0-9]*xy?
Regex Example Solutions

Make a regex for the language of any combination of the letters a and b.

Solution: \( (a \mid b)^* \) or \([ab]^*\)

Make a regex for the language of any number of a’s, followed by an even number of c’s.

Solution: \( a^*(cc)^* \)

Describe the language (with an english sentence) represented by: \( a+[0-9]^*xy? \)

Solution: 1 or more a’s, followed by any number of digits (remember, digit is any number from 0-9), followed by x, and then 0 or 1 y.
Regex Practice

Make a regex for the language of at least 2 b’s, followed by an a, b, or c, followed by two a’s.

Make a regex for the language of any number of a’s, followed by a c or two b’s, followed by an odd number of c’s and then an optional a.

Describe the language (with an english sentence) represented by: a+b?c+b?d*
Make a regex for the language of at least 2 b’s, followed by an a, b, or c, followed by two a’s.

**Solution:** `bb+[abc]aa`

Make a regex for the language of any number of a’s, followed by a c or two b’s, followed by an odd number of c’s and then an optional a.

**Solution:** `a*([^bc])c(c*)a?`

Describe the language (with an english sentence) represented by: `a+b?c+b?d*`

**Solution:** at least one a, followed by an optional b, then at least one c, then an optional b, and then any number of d’s.
Conversions

A lot of ways to do conversions between Regex and DFA/NFA.

Simpler ones can be translated directly. For more complex examples, it is useful to first generate some examples of strings that would be valid and invalid in order to help describe the language (in English). Then convert the language to the required representation as in previous examples.
Regex to DFA Examples

- Draw a DFA for the language \([ab]^*\)
- Draw a DFA for the language \([ac]^*bd+a?\)
Draw a DFA for the language $[ab]^*$

**Solution:**

![DFA Diagram]

- States: $q_0$
- Start state: $q_0$
- Transitions: $(a, b) \rightarrow q_0$

Here, $q_0$ is the start state, and the DFA accepts any string of $a$s and $b$s.
Draw a DFA for the language \([ac]^*bd+a\)?

**Solution:**

![DFA Diagram]

- **Start State:** \(q_0\)
- **Transitions:**
  - \(a,c\) from \(q_0\) to \(q_1\)
  - \(b\) from \(q_1\) to \(q_2\)
  - \(d\) from \(q_2\) to \(q_3\)
  - \(a\) from \(q_3\) back to \(q_0\)
Regex to DFA (and DFA to Regex) Practice

- Draw a DFA for the language $a^*(cc)^*$
- Draw a DFA for the language $(bb|ac)c^*[ad]$
- Write a regex that matches the same language accepted by the following DFA:
Draw a DFA for the language $a^*(cc)^*$

**Solution:**

![DFA Diagram](image-url)
Draw a DFA for the language \((bb|ac)c^*[ad]\)

**Solution:**

[Diagram of the DFA]
Write a regex that matches the same language accepted by the following DFA:

Solution: \((aa|b)b(dd)^*c\)
Assignment reminders

● Submit a `.txt` XOR a `.pdf` for each question
  ○ Do not submit both for the same question!
  ○ You may submit a `.pdf` for one question and a `.txt` for a different question

● Make sure your diagrams and tables are clear and easy to read
  ○ Make sure to leave enough space