Institut for Matematik og Datalogi
Syddansk Universitet

September 3, 2014
JBJ

DM517 – Fall 2014 – Weekly Note 2

Lectures in Week 36:
We covered most of Chapter 1 on regular languages. We will discuss more on Section 1.4 in week 37.

Key points:

- Automata are mathematical models of computers.
- Finite automata (FA) use a constant amount of memory and determine whether a given string is in a certain language. The moves of a deterministic finite automaton (DFA) is completely determined by its initial state and the input string. A DFA M accepts a string w if the walk that starts in the initial state and "spells" w ends in an accepting state of M.
- On the contrary, a nondeterministic finite automaton (NFA) can choose between several alternative moves, and it has the ability to guess the moves that lead to a favorable state. A NFA M accepts a string w if there exists a walk from the start state to one of the accepting (final states) which "spells" exactly w.
- However, NFAs do not have greater computational power than DFAs, as we saw how to convert a NFA M to an equivalent DFA M' (with $L(M') = L(M)$). Note that this conversion is **NOT a polynomial algorithm** since M' may have exponentially many states compared to M!
- A regular expression (RE) is a formal specification of a regular language.
- A language is accepted by some FA if and only if it is generated by some regular expresssion, since an equivalent FA can be constructed from a given RE and vice versa.
- The class of regular languages is closed under concatenation, complement, union, star, and intersection. The following holds:

Let L, L' be regular languages, then each of the following are also regular:

1. $L \cup L'$.
2. The complement $\overline{L}$ of L.
3. $L \cap L'$.
4. $L \setminus L'$.
5. LL'.
6. L^*

To see that $L \cap L'$ is regular, it suffices to observe that $L \cap L' = \overline{\overline{L} \cup \overline{L'}}$.

- **Every finite language is regular**. So every non-regular language contains arbitrarily long strings. In particular, it is infinite.
- Applying the pumping lemma is like a 2 person game between you and an adversary: In order to prove that a language L is NOT regular you proceed as follows:
  - Assume (to reach a contradiction) that $L = L(M)$ for some DFA M and let p be the number of states of M. You may also think of getting p from **the adversary** (who claims that M exists).
  - **You** choose a suitable string $s \in L$ such that $|s| \geq p$.
  - Now **the adversary** must choose strings x, y, z over Σ such that
    1. $s = xyz$
    2. $xy^iz \in L$ for **all** $i \geq 0$
    3. $|y| > 0$ and $|xy| \leq p$.

    By the pumping Lemma (s)he can do so if L is indeed regular.
  - Then **you** choose a suitable $i \geq 0$ and show that $xy^iz \notin L$, contradiction.
  - Since the contradiction arose from the assumption that L was regular, it follows that L is not a regular language.

Lecture September 8, 2014:

- Non regular languages. Section 1.4, continued from last week.
- Context free gramars. Section 2.1.
- Pushdown automata. Section 2.2.

Exercises September 10, 2014:

For those with many subquestions, the instructor will select a subset of these to discuss with you.

- 1.6 page 84 questions a,b,d,f,h,i,k
- 1.7 page 84 questions d,e,g,h
- Solve the following problem: :

  A man is travelling with a wolf (w) and a goat (g). He also brings along a nice big cabbage (c). He encounters a small river which he must cross to continue his travel. Fortunately, there is a small boat at the shore which he can use. However, the boat is so small that the man cannot bring more than himself and exactly one more item along (from $\{w, g, c\}$). The man knows that if left alone with the goat, the wolf will surely eat it and the goat if left alone with the cabbage will also surely eat that. The man's task is hence to device a transportation scheme in which, at any time, at most one item from $\{w, g, c\}$ is in the boat and the result is that they all crossed the river and can continue unharmed.

  (a) Describe a solution to the problem which satisfies the rules of the "game". You may use your answer to (b) to find a solution.

  (b) Consider strings over the alphabet $\Sigma = \{m, w, g, c\}$ and interpret these as follows: The symbol m means that the man crosses the river alone, w means that he brings the wolf etc.

  Design a finite automaton which accepts precisely those strings over Σ which correspond to a transportation sequence where everybody survives and is legal in the sense that the man can only bring an item (e.g. w) back across the river if it was actually on the shore where the boat just left from. For example $gmcg$ is a legal string (it is not a solution) whereas gc is not legal.

- 1.17 on page 86.
- 1.19(a) and 1.21(b) on page 86.
- 1.20 and 1.18(a)-(g) on page 86.