CS 373 Notes

19 February 2012

Contents

Sizes	Examples	Countable?
Finite	{a,b}	yes
Countable Infinite	N,Z, Q	yes
Uncountable Infinity	R, Pow(R)	dno

1 General

1.1 Starting off

1. Alphabet(Σ) = finite non empty set

2. N in this class starts at 0
3. A set X is countably infinite iff $\exists$ a bijection $f : \mathbb{N} \rightarrow X$

1.2 Strings

1. String(w) = sequence of characted in Σ
2. w:{$c_i \in \Sigma$ | 0 <= i <= n}
3. |w| = n = length of the string
4. |w| = 0 $\rightarrow w = \epsilon$

(a) Careful $\sigma \neq \emptyset$

5. Substring subsequence of characters in w
6. Concatination: $w_1 \cdot w_2$
7. Reverse: w^r
8. Palindrum: w = w^r

1.3 Language

1. Language(L) = set of strings
2. Σ^n = {w : |w| = n}
3. Σ^0 = { ϵ }
4. $\Sigma^* = \cup^n{}_{i=0}\ \Sigma^i$, Language of all strings

2 Regular Languages

2.1 Deterministic Finite Automotas

1. Finite state machine (M)
2. Takes a string of inputs
3. 2 types of states
   (a) Accept

(b) Deny

4. There is 1 start state

5. The set of all strings accepted by language of M or L(A)

6. Formal Definition

   (a) a Language A $\in \Sigma^*$ is called regular iff there exists a DFA ,M, s.t. $L(M) = A$

   (b) DFA is a 5 tuple $ M = (Q,Σ, δ, q_0, F)$

      i. Q is a finte set of states
      ii. Σ is a finite alphabet
      iii. $\delta : Q \times \Sigma \rightarrow Q$ is the transition function
      iv. $q_0 \in Q$ is the inital state
      v. $F \subset Q$ is the set of accept states

   (c) L(M) ≡ language of all accepted strings

7. Closure properties/Regular Operations on languages

   (a) A_1 and A_2 are regular
   (b) Union: $A_1 \cup A_2 = A_3$
   (c) Concatenate: A_1 A_2 = A_3
   (d) Star: $A_1{}^* = A_3$

2.2 Non-Deterministic Finite Automotas (NFA)

1. Formal Definition

   (a) M = (Q,Σ,δ,q_0,F)

      i. Q = finite set of states
      ii. Σ is a finate alphabet
      iii. $\delta = Q \times \Sigma_\epsilon \rightarrow Pow(Q)$
         A. $\Sigma_\epsilon = \Sigma \cup \epsilon$
      iv. q_0 = start state
      v. $F \subset Q$

   (b) NFA accepts w If we can write $w = y_1 y_2 ... y_n y_i \in \Sigma_\epsilon$ s.t. there exists a sequence of states path $R = r_0, r_1, ...$

i. $\mathrm{r}_0 = \mathrm{q}_0$
ii. $r_{i+1} \in \delta(r_i, y_{i+1})$ for $i = 0, 1...m-1$
iii. $r_m \in F$

2. Useful Lemma: For all NFA, M, there exists an DFA ,N, s.t. L(m) = L(n)

2.3 Regular Expressions

1. Def: R is a regex over a fixed alphabet iff one of the following is true:
   (a) R = a $\in\Sigma$
   (b) R = σ
   (c) $R = \emptyset$
   (d) $R = R$_1$ \cup$ R$_2$,$ given $R_1 \wedge R_2$ are regex
   (e) $R = R$_1$ R$_2$,$ given $R_1 \wedge R_2$ are regex
   (f) $R = R^*$
2. Order of operations
   (a) star
   (b) concatination
   (c) union
3. Identities
   (a) $a\emptyset = \emptyset$
   (b) $a\sigma = \sigma$
   (c) $\emptyset^* = \sigma$

2.4 Generalized NFA (GNFA)

1. Definition
   (a) Q = set of all states
   (b) $Q^0 = Q - q_start, q_accept$
   (c) The start state has out edges to every $q \in Q - q_start$, and no in edges

(d) The accept state has inedges from every $q \in Q - q_accept$, and no outedges

(e) An edege exists from every $q_1 \in Q^0$ to every $q_2 \in Q^0$ even if $q_1 = q_2$

(f) Every edge is labed with a regex

2. Useful lemma: Any NFA can be written as a GNFA

3. lemma: Given a GNFA, M, with 2 states, the regex between the 2 states describes the language of M

2.5 Pumping Lemma for regular languages

If A is regular, then $\exists p \in \mathbb{N}$ s.t. $\forall s \in A$ for which $|s| >= p, s$ can be written as xyz and satisfy the following conidition:

1. $\forall i >= 0, xy^i z \in A$

2. $|y| > 0$ i.e. $y \neq \sigma$

3. $|xy| <= p$

P is called the "pumping length

2.6 Subsitutions

2.7 Reg Operations (closed under the Reg languages)

1. $A_1 \cup A_2$

2. $A_1 - A_2$

3. $\bar{A}_1 = \Sigma^* - A_1$

4. $A_1 \cap A_2$

5. Symmetric Diff

6. $A_1 A_2$

7. A_1^*

8. A^r

9. Reg langagues are clased under subsitution

10. Reg langagues are clased under homomorphism
11. Reg langagues are clased under inverse homorphism
12. Reg langagues are clased under

3 Non-Regular Languages