Integrated Master's in Informatics and Computing Engineering

Theory of Computation

DFA, NFA, or ε-NFA: $A = (Q, \Sigma, \delta, q_0, F)$

- Example of the extended transition function, supposing the existence of the states *p* and *q*, the string *w*, and transitions *q* to *p* in the automaton with *w*: NFA ε-NFA: $\hat{\delta}(q, w) = \{p\}$; DFA: $\hat{\delta}(q, w) = p$
- Conversion of an FA (finite automaton) to a regular expression using the path construction technique: $R_{ij}^{(k)} = R_{ij}^{(k-1)} + R_{ik}^{(k-1)} (R_{kk}^{(k-1)})^* R_{kj}^{(k-1)}$, where $1 \leq k \leq N$ e $1 \leq i, j \leq N$ (it is assumed that the FA states are enumerated from 1 to N)

PDA (Pushdown Automaton): $A = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$

- Example of a computing trace in a given PDA using instantaneous descriptions: $(q, aw, X\beta) \vdash (p,w,\alpha\beta)$
- Theorem 1: If $(q,x,\alpha) \vdash^* (p,y,\beta)$ then $(q,xw,\alpha\gamma) \vdash^* (p,yw,\beta\gamma)$
- Theorem 2: If $(q,xw,\alpha) \vdash^* (p,yw,\beta)$ then $(q,x,\alpha) \vdash^* (p,y,\beta)$

TM (Turing Machine): $A = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$

- Example of a step in a Turing Machine: $qX_1X_2\ldots X_n \vdash pBYX_2\ldots X_n$ (in this case, the TM is at state *q*, replaces X_1 by Y, goes to the *p* state and moves in tape to the left side)

Cocke-Younger-Kasami Algorithm (CYK): used to test if a string is in a CFL and the CFL is represented by a grammar in CNF.

Regular expression operators:

- * (zero or more occurrences)
- . (concatenation: symbol can be omitted)
- + (or | or ∪)
- Precedence (from the highest to the lowest): *, ., +
- Curve brackets can be used to change the usual precedence order.

Pumping Lemma for Regular Languages:
Given an infinite regular language L. There exists a constant *n* (dependent of L) such that for every string *w* in L with $|w| \geq n$ it is possible to break *w* in 3 substrings w=xyz where:

- $y \neq \varepsilon$
- $|xy| \leq n$
- For every $k \geq 0$, the string xy^kz is also in L.

Chomsky Normal Form (CNF):
All the CFLs without ε (the empty string) has a grammar in the normal form of Chomsky, without useful symbols and in which all the productions are in the form:

- A → BC (A, B, C are variables) or
- A → a (A is a variable and 'a' is a terminal)

Pumping Lemma for regular Languages without context (CFLs):
Consider L an infinite CFL. Then exists a constant *n* such that, for any string *z* in L with $|z| \geq n$ we can break *z* in z=uvwxy such that:

- $|vwx| \leq n$
- $vx \neq \varepsilon$
- For every $i \geq 0$, $uv^iwx^iy \in L$