Formal Languages & Automata Theory

Lecture 1 :

Introduction

Dr. Ebtsam Abdelhakam Sayed
Faculty of Computers and Information
Minia University

Overall Aims of Course:

By the end of the course the students will be able to:

1. Understand the concepts of the Formal Languages & Automata Theory.
2. Define and understand the Automata Theory aspects.
3. Design an automata diagrams.
4. Know common applications for Formal Languages & Automata Theory.
5. Choose the appropriate Formal Languages & Automata Theory for modeling a given problem.

Contents:

Lec #	Topic	No. of hours	Lecture	Tutorial/ Practical
Lec.1	Introduction	3	2	2
Lec.2	FA - Deterministic finite Automata (DFA)	3	2	2
Lec.3	Nondeterministic Finite Automata (FNA)	3	2	2
Lec. 4	Conversion from NFA with ε to NFA without ε	3	2	2
Lec. 5	Conversion from NFA to DFA	3	2	2
Lec. 6	Conversion from NFA with ε to DFA	3	2	2
Lec. 7	Conversions and Equivalence	3	2	2
Lec. 8	Minimization of FSM	3	2	2
Lec. 9	Finite Automata with Output (Task)	3	2	2
Lec. 10	Regular Expression	3	2	2
Lec. 11	Grammar Formalism	3	2	2
Lec. 12	Pushdown Automata	3	2	2
Lec. 13	Turing machines (Task)	3	2	2
Lec. 14	Pumping Lemma (Task)	3	2	2
Total hours		42	28	14

Evaluation:

Course degree	100 Degree
Final-term Examination	60 %
Mid Term Examination	10 %
Practical Examination	10 %
Oral Examination	10 %
Semester Work	10 %
Total	100%

1.1 Introduction

Formal languages and automata theory is based on mathematical computations. These computations are used to represent various mathematical models. In this subject, we will study many interesting models such as finite automata, push down automata and Turing machines. We will also discuss regular languages, non-regular languages. Context free languages. This subject is a fundamental subject, and it is very close to the subjects like compilers design operating system, system software and pattern recognition system.

The automata theory is a base of this subject. Automata theory is a theory of models. Working of every process can be represented by means of models. The model can be a theoretical or mathematical model. The model helps in representing the concept of every activity. In this chapter, we will discuss all the fundamentals of automata theory and those and those re strings, languages operations on the languages. In the latter part of the chapter we will understand the concept of finite state machine, transition diagrams, and language recognizers.

1.2 Basic Concepts

The automata theory has a basic unity called set. The set is used to represent the mathematical model. Hence let us discuss basics of set theory.

1.2.1 Set

Set is defined as collection of objects. $A = \{0,1,2,3,4\}$

If 'a' is an element of set A then we say that a $\in$ A and if a is not an element of A then we say that a $\notin$ A.

Empty set: **The set having no element in it is called empty set. It is denoted by A = {}**

and it can be written as ϕ ***(phi).***

Null string: **the null element is denoted by ε or** $\wedge$ **character. Null element means no**

value character. But ε does mean ϕ

Power set: **The power set is a set of all the subsets of its elements.**

For example: A = {1, 2, 3}

Then power set: Q = $\{\phi, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{3,2\}, \{2,3\}, \{1,2,3\}\}$

The number of elements is always equal to 2^n where n is number of elements in original set.

As in set A there are 3 elements. Therefore, in power set Q there are 2^3=8 elements.

1.2.2. Operations on set

Various operations that can be carried out on set are -

i) Union

ii) Intersection

iii) Difference

iv) Complement

Self study

- ❑ Cartesian Product of Two Sets
- ❑ Cardinality of Sets
- ❑ Relations
- ❑ Formal Proofs
  - ✓ Proof by Contradiction
  - ✓ Proof by Counter Example
  - ✓ Inductive Proofs

Introduction to Defining Language

Alphabet : An alphabet (∑) is a finite collection of symbols.

The example of alphabet is,

S = {a, b, c, … z}

The elements a, b, … Z are the alphabets.

W = {0, 1}

Here 0 and 1 are alphabets, denoted by W.

String: **String is finite collection of symbols from alphabet.**

For example, if ∑ - {a, b} then various strings that can be formed ∑ are {ab, ab, aa, aaa, bb, ba, aba …} An infinite number of strings can be generated from this set.

- The empty string can be denoted by ε.
- The prefix of any string is any number of leading symbols of that string and suffix is any number of trailing.

Language: The language is a collection of appropriate strings the language is defied using an input the input set is typically denoted by letter $\sum$.

For example: $\sum = \{\varepsilon, 0, 00, 000, \dots\}$, Here the language L is defined as any number of Zeros.

Note that language is formed by appropriate strings and strings are formed by alphabets.

Operations on string

Various operations that can be carried out on strings are.

1- Concatenation:

In this operation two strings are combined to form a single string.

For example, x = (white) y = (house).

Then the concatenated string will be xy = (white house).

2- Transpose:

The transpose of operation is also called reverse operation.

For example, $(xa)^T = a(x)^T$ if $(ababbb)^T = (bbbaba)$

3- Palindrome:

Palindrome of string is a property of a string in which string can be read same from left to right as well as from right to left. For example, the string " MadaM " is palindrome because it is same if we read it from left to right or from right to left.

Operations on language

Language is collection of strings. Hence operations that can be carried out on strings are the operations that can be carried out on language.

If L_1, and L_2, are two languages then,

i) Union of two language is denoted as $L_1 \cup L_2$.

ii) Concatenation of two languages is denoted by $L_1 L_2$.

iii) Intersection of two languages is denoted by $L_1 \cap L_2$.

iv) Difference of two languages is denoted by $L_1 - L_2$.

Example 1.1: Consider the string X = 110 and y = 0110 then found.

(i) xy (ii) yx (iii) X^2 (iv) εy

Solution: let x = 110 and y = 0110 then

(i) xy is the concatenation operation. Hence, we will concatenate the two strings from x y respectively. xy = 1100110

(ii) yx suggest the concatenation if string from set y with string from x hence: yx=0110110.

(iii) x^2 = x.x Hence concatenate the string from x with the string from set x itself. Hence x^2 = 110110.

(iv) ε y means the string belonging to set y which is 0110.

Example 1.2: Describe the following language over the input set A = {a, b}. i) L1 = (a, ab, ab^2) ii) L2 = $\{a^n b^n \mid n \geq 1\}$

Solution: The language is defined over the set A = (a, b).

i) L1 consists of all the strings with letter a followed by any number of b's.
ii) L2 consists of all the strings having equal number of a’s and equal number of b's such that all the a's are followed by all the b's.

Kleene Closure

The Kleene closure is also called star closure. This is set of strings of any length (including null string ε). Each string is obtained from input set $\sum$ the Kleene star of the empty language ϕ is the empty string ε.

Example 1.3: Let $\sum = \{a, b\}$ obtain $\sum^{+} = \sum^{0} \cup \sum^{1} \cup \sum^{2} \cup \sum^{3} \ldots\ldots\ldots$
Solution: $\sum^{*} = \{\varepsilon, a, b, aa, bb, ab, ba, aba, aaa, bbb, baba, abaab \ldots\ldots\}$ is a set of strings of any length. The positive closure $\sum^{+}$ can be defined as $\sum 1 \cup \sum^{2} \cup \sum^{3}$ that means it consists of all the strings of any length except a null string.

Example 1.4: Let $\sum$ {a, b} obtain $\sum^{*} = \sum^{1} \cup \sum^{2} \cup \sum^{3} \ldots\ldots$
Solution: $\sum^{*} = (a, b, aa, bb, ab, ba, aba, aaa, bbb, baba, abaab, \ldots)$ is a set of strings of any length except null string (epsilon).
Note that $\sum^{*} = \sum^{+} + \varepsilon$.

Any Questions?