Languages and Grammar

Lecturer: Manar Elkady, Ph.D

Prepared by: Prof. Akram Salah

Presented by: Manar Elkady, PhD

BASIC CONCEPTS

NUMBERS AND DIGITS

- What is the difference between number and digit
- ما الفرق بين العدد و الرقم (الاعداد و الارقام)

NUMBERS AND DIGITS

- Digits (in decimal system) are a set of symbols
  - {0, 1, 2, 3, 4, 5, 6, 7, 8, 9} (only)
  - Symbols, don’t have meaning
- Numbers have meaning
  - Number of students in class 32
  - Number of students at FCAI 3587
  - Number of faculty members 423
- A number is *represented* as a string of digits

NUMBERS AND DIGITS

- *The representation is a string, the meaning has to be interpreted.*
- The set N (natural numbers) is represented as a string of digits {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
  - 32, 3587, 423
- The set Z (Integers) is represented as a string of characters {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -}
  - 46, -18, 0

NUMBERS AND DIGITS

- The set Q (Rational numbers) is represented as a string of characters {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, /}
  - 5/8, 3/11, -4/3
- The set R (real numbers, or floating point) is represented as a string of characters {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, .}
  - 25.67, 3.14, -118.0
    - What is the difference between 18.0 and 18

NUMBERS AND DIGITS

- So a number represents a quantity in real life and is written (expressed) as a string of digits.
- In the language of numbers, each number represented as a string from a set of characters coming from an alphabet.
  - In decimal systems digits are {0, 1, …, 9}
  - In binary systems digits are {0, 1}
  - In octal systems digits are {0, 1, 2, 3, 4, 5, 6, 7}

NUMBERS AND DIGITS

- When I write a number I write a string.
- The meaning is understood (interpreted) in mind.
- For computers, the string is converted to a number through a process then stored as a value.

INTRODUCTION

- To generalize, writing is string expressions that are later to be interpreted.
- Strings come from a set of primitive symbols.
  - English expressions come from the set of 26 alphabet + set of digits + special characters
    - Words in English are strings.
    - A sentence is a string of words
    - Paragraph is composed of sentences.
  - A line of text is a string.

INTRODUCTION

- Arabic expressions come as strings from 28 alphabet, set of digits, special characters, and punctuation marks.(التشكيل) علامات الضبط
  - Sentences
  - Paragraphs
  - ....

INTRODUCTION

- So, communication, in writing, of a thought (the thought in your mind) *expressed as string* of text, then transferred to another person who *reads* the string and *understands* the text.
- As we communicate with computers, we express what we want as a string of text, then it will be interpreted internally to a meaning.
  - Could be a command, data, or instruction.

BASIC CONCEPTS

- In our course, formal methods and theory of computer we will discuss three major branches of concepts.
  - Languages
    - Forming strings for communication
  - Grammar
    - Rules of valid expressions
  - Automata (Automaton)
    - “Machines” that interpret expressions

LANGUAGES

LANGUAGES

- A body of communicative symbols, whether it is made up of words, sounds, gestures, facial expressions, or visual signals that by long use by the population of a wide spread territory has became whereby the ideas or feelings of the individual members of that population are communicated or expressed.
- A system suitable for the expression of certain ideas, facts, concepts, including a set of symbols and rules for their manipulation.

LANGUAGES

- Alphabet Σ : finite, non empty set of symbols
- String: finite sequence of symbols from Σ
  - Σ = {a, b},
  - abab, aaabbba are strings on Σ
  - *w* = abaaa called a word
  - A language is a set of words, each is a string, defined over Σ

LANGUAGES

- $w = a_1a_2 \ldots a_n$ and $v = b_1b_2 \ldots b_m$
- Concatenate:
  - $wv = a_1a_2 .. a_nb_1b_2 ..b_m$
- Reverse:
  - $w^R = a_n .. a_2\, a_1$
- If w = *vu*,
  - *v prefix* of w
  - *u suffix* of w

LANGUAGES

- Length of a string w , $|w|$
  - The number of symbols in w
- Empty string Λ a string with no symbols
  - $|\Lambda| = 0$
  - $\Lambda w = w\Lambda = w$, for all w
  - $|uv| = |u| + |v|$

LANGUAGES

- w is a string, w^n is concatenating w with itself n times.
    - $(ab)^2$ is abab
  - $w^0 = \Lambda$
- If Σ is an alphabet,
  - Σ^* is the set of strings concatenating 0 0r more elements of Σ.
  - Σ^+ is the set of strings concatenating 1 or more symbols.
  - $\Sigma^+ = \Sigma^* - \{\Lambda\}$.
  - $\Sigma^* = \Sigma^+ \cup \{\Lambda\}$

LANGUAGES

- A language, L, defined over an alphabet Σ, is a subset of Σ^*
- Any string w in a language L is called a word or a sentence.
- We can define more than one language over the same Σ.
  - Complement of a language $L^c = \Sigma^* - L$.
  - Concatenation of two languages

GRAMMARS

GRAMMARS

- A grammar for a language tells us whether a particular sentence is well formed or not.
- A set of rules.
- We need a precise non-ambiguous notation to describe the rules.
  - The rules simply tells us which subset of L is accepted as a “legal” expression

GRAMMARS

English sentence

<sentence> -> <noun_phrase><predicate>,

<noun_phrase> -> <articale><noun>,

<predicate> -> <verb>,

<article> {“a”, “the”}

<noun> {“boy”, “dog”}

<verb> {“walks”, “runs”}

“a boy runs” | Properly formed

“the dog walks” |

GRAMMARS

- A grammar G is a quadruple

$$G = (V, T, S, P)$$

Where:

V finite set of objects called ***variables***

T finite set of objects called ***terminal*** symbols

$S \varepsilon V$ is a special symbols called ***start*** variable

P is a finite set of ***productions*** *or* ***production rules***

V & T are non-empty and disjoint

GRAMMARS

- Production rules are the heart of a grammar
- They specify how the grammar transforms one string to another
- They define a language associated with a grammar G
- Production rule in the form:

$$x \to y$$

$$x \in (V \cup T)^{+} \quad \& \quad y \in (V \cup T)^{*}$$

GRAMMARS

- Given a string

  $w = uxv,$

  rule x -> y is *applicable* to w

  $z = uyv.$

  written as $w \Rightarrow z$,

  w derives z or *z is driven from w*

GRAMMARS

- If $w_1 => w_2 => \ldots w_n$,

  we write $w_1 =^* > w_n$
- "*" indicates 0 or more steps to derive w_n from w_1

  $w =^*> w$

  $w =^+> v$

GRAMMARS

- Applying productions in different order, a grammar generates many strings
- $G = (V, T, S, P)$,

  $$L(G) = \{ w \,\varepsilon\, T^* : S =^{*}> w\}$$

  - $L(G)$ *is Language generated by* G
  - $S => w_1 => w_2 => \ldots => w_n => w$ *is* ***derivation*** *of* w
  - $S, w_1, w_2, \ldots, w_n$, ***sentential forms*** *of the derivation*

GRAMMARS

- Example:
- Given G = ({S}, {a, b}, S, P)

  Where P $S \to aSb$,

  $S \to \Lambda$

  Then $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$

  $S \Rightarrow aSb \Rightarrow ab$

GRAMMARS

- Given $G = (\{S\}, \{a, b\}, S, P)$

  Where P $S \to aSb,$

  $S \to \Lambda$

  Then $S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$

  $L(G) = \{a^n b^n : n \geq 0\}$

GRAMMARS

- Example:
- Find a grammar that generates

  $L= \{a^n b^{n+1} : n \geq 0\}$

GRAMMARS

- G ({S, A}, {a,b}, S, P)

  where P S -> Ab,

  A -> aAb,

  A -> Λ.
- We can write P

  S -> Ab,

  A -> aAb | Λ

AUTOMATA