Universidad Carlos III de Madrid
www.uc3m.es

AUTOMATA THEORY AND FORMAL LANGUAGES

UNIT 2: AUTOMATA THEORY

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Universidad Carlos III de Madrid
www.uc3m.es

OUTLINE

- Introduction to Automata Theory

- Formal Languages: Introduction and Definitions
  - Definitions
    - Symbols (of a Language)
    - Alphabet (of a Language)
    - Word (or string); Length of a Word; Empty Word
    - Universe of an Alphabet

  - Operations with Words. Operations with Languages

OUTLINE

- **Introduction to Automata Theory**

- Formal Languages: Introduction and Definitions
  - Definitions
    - Symbols (of a Language)
    - Alphabet (of a Language)
    - Word (or string); Length of a Word; Empty Word
    - Universe of an Alphabet
  - Operations with Words. Operations with Languages

Introduction and definitions

- In this subject we are interested in knowing whether a problem is computable or not.

  In addition…
  time required,
  memory required, and
  computation model to be used.

Introduction and definitions

- Which is the meaning of **computation**?
- Automata Theory is focused on the computation itself, not in the detailed definition of input and devices.

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Mathematical Model of an Automaton

Oxford English Dictionary

Automaton.

(Latin, self-operating machine, from Greek, from neuter of automatos, self-acting; see automatic.).

1. Instrument or device which includes an internal mechanism to facilitate an automatic movement.

2. Machine made in imitation of a human being (appearance and movements).

3. A person who acts mechanically or leads a routine monotonous life.

Mathematical Model of an Automaton

Automaton:

Mathematical model defined for computation.

Automata:

Abstraction of any type of computer and/or programming language.

Set of basic elements (Inputs, States, Transitions, Outputs and auxiliary elements)

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Types of Automata

Finite Automata (and sequential machines)

Push-Down Automata

Linear-Bounded Automata

Turing Machine

Cellular Automata

Artificial Neural Networks

…

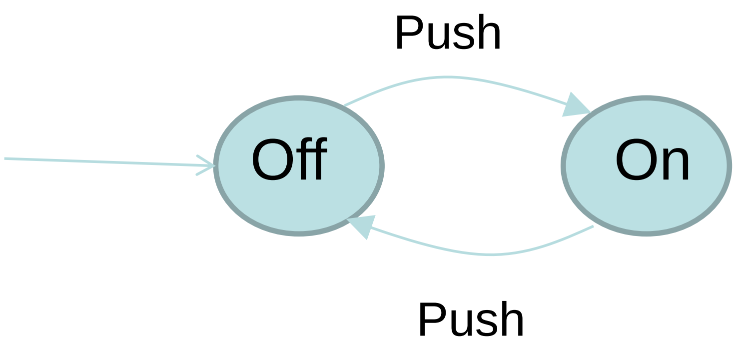
capacity of calculation

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Automata and Algorithms

Every automaton can be transformed into an algorithm and viceversa.

Finite Automaton:



David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Discrete, Continuous and Hybrid Automata

Criteria: Number of inputs.

Usually DISCRETE:

- Finite Automata (and sequential machines)
- Push-Down Automata
- Turing Machines

DISCRETE, CONTINUOUS AND/OR HYBRIDS:

- Cellular Automata
- Artificial Neural Networks

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

ROBOTS BEHAVIOUR IN RoboSoccer

- Finite state automata applied to robot soccer (Peter van de Ven)

http://www.google.es/rdr?sa=t&source=web&cd=9&ved=0CEwQFjAI&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload%3Fdoi%3D10.1.1.125.1893%26rep%3Drep1%26type%3Dpdf&rct=j&q=robotics%20finite%20automata&ei=BxOvTJafAoWclgf0uInXBA&usg=AFQjCNGPqCsTw6wBHgP44uPpTbFENwERtQ&cad=rjt

GAME OF THE LIFE

- Example of a cellular automaton, designed by the British mathematician John Horton Conway in 1970.
- Transitions depend on the number of alive neighboring cells:
  - A dead cell that has exactly 3 alive neighboring cells will be alive in the following turn.
  - An alive cell with 2 or 3 alive neighboring cells follows alive, in another case it dies or remains dead (by “isolation” or “overpopulation”).
- http://www.youtube.com/watch?v=XcuBvj0pw-E
- http://www.youtube.com/watch?v=FdMzngWchDk
- http://www.youtube.com/watch?v=k2IZ1qsx4CM&NR=1

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

OUTLINE

- Introduction to Automata Theory

- **Formal Languages: Introduction and Definitions**
  - **Definitions**
    - **Symbols (of a Language)**
    - **Alphabet (of a Language)**
    - **Word (or string); Length of a Word; Empty Word**
    - **Universe of an Alphabet**

  - Operations with Words. Operations with Languages

Introduction and Definitions

- **Symbol:** an abstract entity with indivisible nature.
  - No formal definition (“point” in geometry).
  - Letters, digits, characters, etc.
  - It is possible to find symbols that consist of several characters, e.g.: IF, THEN, ELSE, ...
- **Alphabet (Σ):** Finite set not empty of symbols.
  - Notation: Let “*a*” be a letter and let Σ be an alphabet:

    a belongs to this alphabet $\Rightarrow a \in \Sigma$
  - Examples:

    $\Sigma_1 = \{A, B, C, ..., Z\}$

    $\Sigma_2 = \{0, 1\}$

    $\Sigma_3 = \{IF, THEN, ELSE, BEGIN, END\}$

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Introduction and Definitions

- **Word, string:** Finite sequence of symbols drawn from an alphabet (sentence and string are also use as synonyms).
  - Examples:

    Words over Σ_1 → HOME, BALL, etc.

    Words over Σ_2 → 00011101

    Words over Σ_3 → IFTHENELSEEND

Note: Words are represented by lowercase by the end of the alphabet (x, y, z), e.g., x= HOME, y= IFTHENELSEEND

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Introduction and Definitions

- **Length of a word:** Number of symbols composing the word.
  - The length of a word *x* is represented by $|x|$

  Examples:

  $|x| = |\text{HOME}| = 4$

  $|y| = |\text{IFTHENELSEEND}| = 13$ → FALSE, due to the definition of the alphabet, the length is 4

- **Empty word λ:** Word of length 0.
  - It is represented by λ, $|\lambda| = 0$
  - In every alphabet, it is possible to construct λ
  - Utility: it is the neutral element in many operations with words and languages.

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Introduction and Definitions

- **Universe of an alphabet, W(Σ):** All the words that can be created using the symbols of the alphabet Σ

  - It is also called Universal Language of Σ.
  - It is represented by W(Σ).
  - It is an infinite set.
  - E.g., given the alphabet Σ_4 = {A},

    $W(\Sigma_4)$ = {λ, A, AA, AAA, ...} with a ∝ number of words

COROLLARY:

$\forall\ \Sigma,\ \lambda \in W(\Sigma) \Rightarrow$ The empty word is included in every Universal Language of every alphabet.

Universidad Carlos III de Madrid
www.uc3m.es

OUTLINE

- Introduction to Automata Theory

- Formal Languages: Introduction and Definitions
  - Definitions
    - Symbols (of a Language)
    - Alphabet (of a Language)
    - Word (or string); Length of a Word; Empty Word
    - Universe of an Alphabet

  - **Operations with Words. Operations with Languages**

Formal Languages. Operations with Words

- **Operations with words: on a given speech universe words.**
  1. Words concatenation.
  2. Word power.
  3. Word reflection.

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Formal Languages. Operations with Words

- **Operations with words:**
  1. **Word Concatenation:** given two words *x*, *y* that fulfill $x \in W(\Sigma)$, $y \in W(\Sigma)$, and $|x| = i = |x_1x_2...x_i|$ and $|y| = j = |y_1y_2...y_j|$,

     The concatenation of *x* with *y* is:

     $x \cdot y = xy = x_1x_2...x_i\, y_1y_2...y_i = \mathbf{z}$, where $z \in W(\Sigma)$

     (symbols of x followed by symbols of y)

Properties:
- Closed operation.
- Asociative.
- With neutral element.
- Not conmutative.

Definitions:
- Head.
- Cue.
- Lenght of the word.

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Formal Languages. Operations with Words

- **Operations with words:**
  - **3. Powers of an Alphabet:** reduction of the concatenation to cases relating to the same word.
    - *i-th power* of a word is the result of the concatenation of this word with itself *i* times
    - Associate property ⇒ the order is not necessary
    - $x^i = x \,.\, x \,.\, x \,. \,....x$ *i* times
    - $|x^i| = i \,.\, |x|$
    - It fulfills:

      $x^1 = x$

      $x^{1+i} = x \,.\, x^i = x^i \,.\, x$ (i>0)

      $x^{j+i} = x^j \,.\, x^i = x^i \,.\, x^j$ (i, j>0)
    - Due to its definition, $x^0 = \lambda$
      - (i, j>=0)

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Formal Languages. Operations with Words

- **Operations with words:**

4. Word reflection: Be x= $A_1A_2A_3...A_n$ a word, the reflected word of x is

$$x^{-1} = A_n... A_3A_2A_1$$

  - Word with the same symbols in inverse order.
  - $|x^{-1}| = |x|$

Formal Languages. Language

- **Language, L:** A language over the alphabet Σ is:
  - Every subset of the universal language of Σ, $L \subset W(\Sigma)$
  - Every subset of words over a specific Σ (generated from the alphabet of Σ)

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

Formal Languages. Language

- **Language, L:**
  - Special languages:
    - **ϕ** = Empty language, $\phi \subset W(\Sigma)$
    - **$\{\lambda\}$** = Language of the empty word.

      they are differenced due to the number of words (cardinality):

      $$C(\phi) = 0 \text{ and } C(\{\lambda\})=1$$

      ϕ y $\{\lambda\}$ are languages generated over every alphabet.
    - An alphabet is one of the languages generated by itself:

      $\Sigma \subset$ **W(Σ)**, e.g. Chinese
    - There are infinite languages associated to an alphabet.

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:** over a given alphabet

  1. Union of languages.
  2. Languages concatenation.
  3. Power of a language.
  4. Positive closure of a language.
  5. Iteration or closure of a language.
  6. Reflection of languages

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:** over a given alphabet

1. **Union of languages**

   - Let L_1 y L_2 be two languages defined using the same alphabet, $L_1, L_2 \subset W(\Sigma)$, the union of these two languages, L_1, L_2 is represented by $L_1 \cup L_2$ and defined by

     $$\mathbf{L_1 \cup L_2 = \{x \mid x \in L_1 \text{ OR } x \in L_2\}}$$

   - Set of words from each one of the languages (equivalent to the plus operation).

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:**
  1. **Union of languages:**

     Properties:

     - Closed operation
     - Associative property $(L_1 + L_2) + L_3 = L_1 + (L_2 + L_3)$
     - With neutral element $\phi + L = L$
     - Conmutative $L_1 + L_2 = L_2 + L_1$
     - Idempotent $L + L = L$

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:** over a given alphabet

2. Concatenation of languages

- Let L_1 and L_2 be two languages defined given the same alphabet, $L_1, L_2 \subset W(\Sigma)$, the **concatenation or product** of two languages, L_1 and L_2 is represented by $L_1 . L_2$ and defined by the language:

$$\mathbf{L_1.L_2 = \{xy \mid x \in L_1 \text{ AND } y \in L_2\}}$$

- Set of words that consists of the concatenation of the words of L_1 with the words of L_2
- Valid definition for languages with almost one element.
- In the case of the empty language: $\phi \,.\, L = L.\, \phi = \phi$

- **Operations with languages:**

  2. Concatenation of languages

  Properties:

  - Closed operation
  - Associative property
  - With a neutral element
  - Distributive property with regard the union

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:**

  4. Powers of a language

  - Particular case of the concatenation for only one language.
  - *i-th* power of a language = result of concatenating this language with itself *i* times.
  - Associative property $\Rightarrow$ it is not needed to specify the order
  - $L^i = L . L . L . .... L$ *i* times
  - Given that $L^1 = L$,

    then:

    $L^{1+i} = L.L^i = L^i.L$ (i>0)

    $L^{j+i} = L^i.L^j$ (i, j >0)

  - Defining $L^0 = \{\lambda\}$ → (i≥0) (i, j ≥ 0)

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:** over a given alphabet

5. Positive Closure of a language

- It is denoted by L^+ and it is the language that consists of joining a language L with all its powers except L^0

$$L^+ = \bigcup_{i=1}^{\infty} L^i$$

- λ is not included in the positive closure if $\lambda \notin L$
- As Σ is a language over Σ, the positive closure of Σ is:

$$\Sigma^+ = \bigcup_{i=1}^{\infty} \Sigma^i = W(\Sigma) - \{\lambda\}$$

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:**

  6. Iteration, closure of a language

  - It is defined by L^* and it is the language that consists of joining the language L and all its possible powers.
  - *** is the Kleene unitary operator**

$$L^* = \bigcup_{i=0}^{\infty} L^i$$

λ Is included in every closure,

  - Properties:
    - **$L^* = L^+ \cup \{\lambda\}$**
    - **$L^+ = L^* \cdot L = L \cdot L^*$**
  - Given that Σ is a language over Σ, we get the universal language: **$\Sigma^* = W(\Sigma)$** → **The universal language is Σ^***

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es

- **Operations with languages:**
  - **7. Reflection of languages**
    - The reflected or inverse language of L is represented by L^{-1} and defined by the language:

$$L^{-1} = \{ x^{-1} \mid x \in L \}$$

    - It is the language that consists of every reflected word of L.

David Griol Barres - Computer Science Department – UC3M - dgriol@inf.uc3m.es