CSE322

Formal Languages and Automation Theory

Lecture #0

Course Details

- CSE322
- LTP – 3 0 0 [Three lectures/week]
- Credit- 3

Vision

To be a globally recognized school through excellence in teaching, learning and research for creating Computer Science professionals, leaders and entrepreneurs of future contributing to society and industry for sustainable growth.

Mission

- To build computational skills through hands-on and practice-based learning with measurable outcomes.
- To establish a strong connect with industry for in-demand technology driven curriculum.
- To build the infrastructure for meaningful research around societal problems.
- To nurture future leaders through research-infused education and lifelong learning.
- To create smart and ethical professionals and entrepreneurs who are recognized globally

Course Outcomes

- Understand Concepts and Abstractions for Automata as a Fundamental Computational Model
- Understand algebraic formalisms of languages such asregular expressions, context-free grammar.
- Compare different types of Grammars and design context free grammars for formal languages
- Analyze the properties and structure of context-free languages
- Understand the construction of Push Down Automata, including closure properties and their relationship with parsing techniques.
- Understand algorithms and computability through the lens of Turing machines and relationship between various computational models.

Program Outcomes

- PO1::Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

- PO2::Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

- PO3::Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

- PO4::Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

Program Outcomes

- PO5::Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

- PO6::Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

- PO7::Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

- PO8::Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Program Outcomes

- PO9::Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings

- PO10::Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11::Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

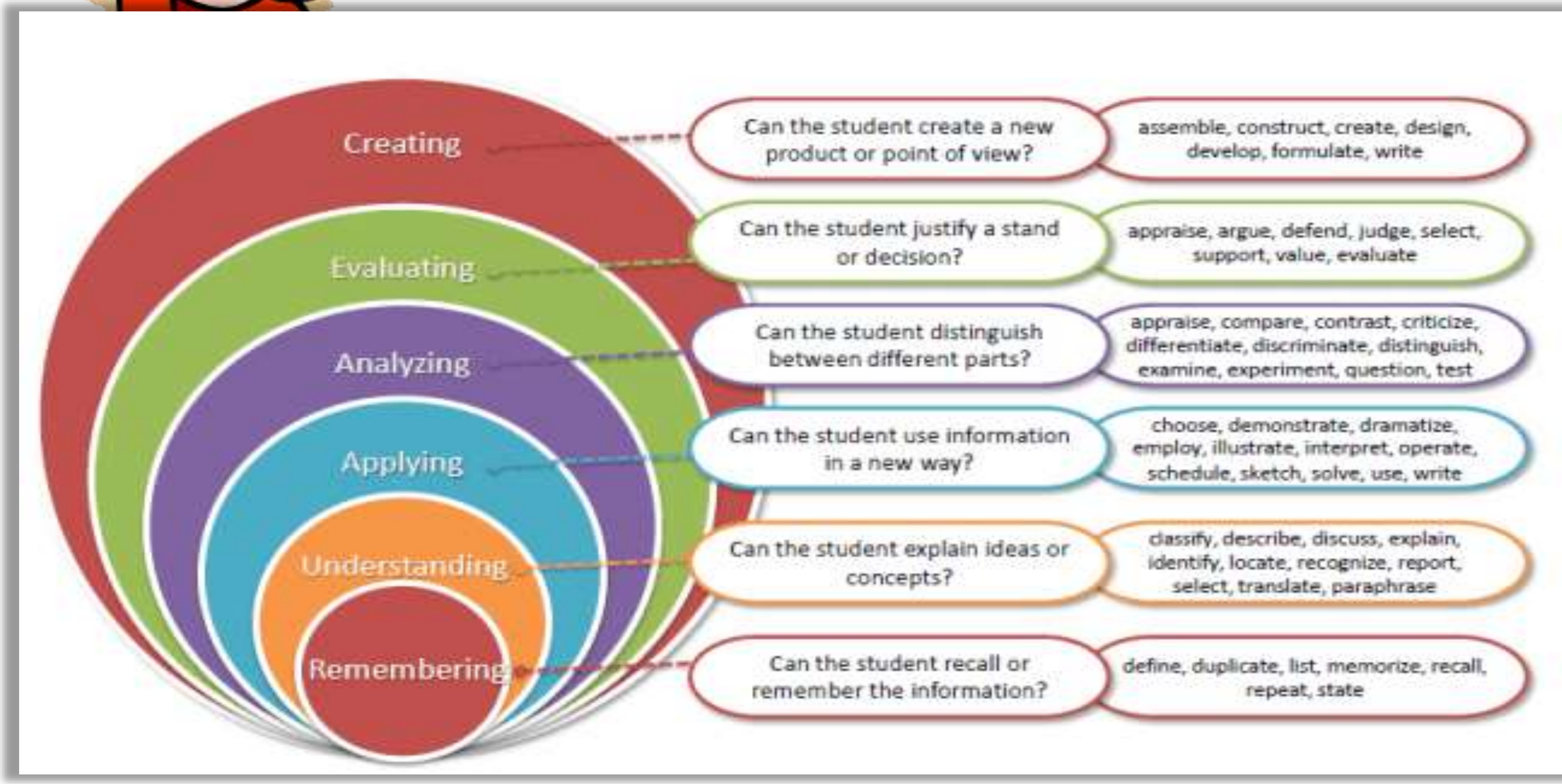
PO12::Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Program Specific Outcome

PSO1: Apply acquired skills in software engineering, networking, security, databases, intelligent systems, cloud computing and operating systems to adapt and deploy innovative software solutions for diverse applications.

PSO2: Apply diverse IT skills to design, develop, and evaluate innovative solutions for business environments, considering risks, and utilizing interdisciplinary knowledge for efficient real-time projects benefiting society.

Revised Bloom's Taxonomy



Course Contents

Unit 1 FINITE AUTOMATA: Definition and Description of a Finite Automaton, Deterministic and Non deterministic Finite State Machines, Transition Systems and Properties of Transition Functions, Acceptability of a String by a Finite Automaton, The Equivalence of DFA and NDFA, Mealy and Moore Machines, Minimization of Finite Automata, Basics of Strings and Alphabets, Transition Graph and Properties of Transition Functions, Regular Languages, The Equivalence of Deterministic and Non deterministic Finite Automata

Unit 2 REGULAR EXPRESSIONS AND REGULAR SETS : Regular Expressions and Identities for Regular Expressions, Finite Automata and Regular Expressions: Transition System Containing null moves, NDFA with null moves and Regular Expressions, Conversion of Non-deterministic Systems to Deterministic Systems, Algebraic Methods using Arden's Theorem, Construction of Finite Automata Equivalent to a Regular Expression, Equivalence of Two Finite Automata and Two Regular Expressions, Closure Properties of Regular Sets, Pumping Lemma for Regular Sets and its Application, Equivalence between regular languages: Construction of Finite Automata Equivalent to a Regular Expression, Properties of Regular Languages, Non-deterministic Finite Automata with Null Moves and Regular Expressions, Myhill-Nerode Theorem

Course Contents

Unit 3 FORMAL LANGUAGES : **Derivations and the Language Generated by a Grammar, Definition of a Grammar, Chomsky Classification of Languages, Languages and their Relation, Recursive and Recursively Enumerable Sets, Languages and Automata, Chomsky hierarchy of Languages REGULAR GRAMMARS**
: Regular Sets and Regular Grammars, Converting Regular Expressions to Regular Grammars, Converting Regular Grammars to Regular Expressions

Unit 4 CONTEXT- FREE LANGUAGES : **Ambiguity in CFG, Leftmost and rightmost derivations, Language of a CFG, Sentential forms, Applications of CFG, Pumping Lemma for CFG, Derivations Generated by a Grammar, Construction of Reduced Grammars, Elimination of null and unit productions, Normal Forms for CFG: Chomsky Normal Form SIMPLIFICATION OF CONTEXT- FREE GRAMMARS :**
Construction of Reduced Grammars, Greibach Normal Form

Course Contents

<u>Unit 5: PUSHDOWN AUTOMATA AND PARSING :</u> Description and Model of Pushdown Automata, Representation of PDA, Acceptance by PDA, Pushdown Automata: NDPDA and DPDA, Context free languages and PDA, Pushdown Automata and Context-Free Languages, Comparison of deterministic and non-deterministic versions, closure properties, LL (k) Grammars and its Properties, LR(k) Grammars and its Properties, PARSING: Top-Down and Bottom-Up Parsing

<u>Unit 6: TURING MACHINES AND COMPLEXITY :</u> Turing Machine Model, Representation of Turing Machines, Design of Turing Machines, The Model of Linear Bounded Automaton, Power of LBA, Variations of TM, Non-Deterministic Turing Machines, Halting Problem of Turing Machine, Post Correspondence Problem, Basic Concepts of Computability, Decidable and Undecidable languages, RECURSIVELY ENUMERABLE LANGUAGE, Computational Complexity: Measuring Time & Space Complexity, Power of Linear Bounded Automaton, Variations of Turing Machine, Cellular automaton

Course Assessment Model

1. >=90% -- 5 marks
2. >=85% and <90% -- 4 marks

Each CA would be of 30 marks.
Best 2 would be taken at the end

MTE would be of 40 marks

ETE would be of 70 marks and it would be prorated to 50 at the end

- 5
-) 20
- MTE 25
- ETE 50
- **Total** **100**

Course Assessment Model

Marks break up

➢ Attendance	5
➢ CA(Best 2out of 3)	25
➢ MTE	20
➢ ETE	50
Total	**100**

Detail of Academic Tasks

- AT1:Test1 (MCQ based) Lecture #11
  (Before MTE)

- AT2:Test2(MCQ based) Lecture #19
  (Before MTE)

- AT3:Test3(MCQ based) Lecture #33
  (After MTE)

Cohort

Government jobs
Higher studies

SkillSet

Analytical Thinking
Problem-Solving
Programming Skills

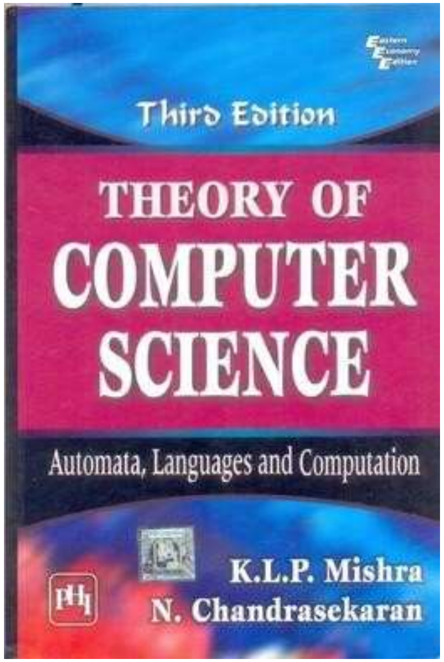
Blended Learning

- Hands-On Exercises: Implement hands-on sessions where students use software tools to model and simulate automata and formal languages.
- Software: Tools like JFLAP can be used for practical exercises.

Text /Reference Book

Text Book

- Theory of Computer Science: Automata, Languages and Computation
  - Author: KLP Mishra and N. Chandrasekaran



MOOCS Details

Course Name :Introduction to Automata, Languages and Computation

Details:

- Category :Computer Science and Engineering
- Credit Points :3

Link:https://onlinecourses.nptel.ac.in/noc24_cs71/preview

Organization :- Swayam

Academic Benefits : All ATs will be exempted

OER(Open Education Resource)

Course Code	Course Title	Unit mapped	Broad topic	OER Type	Title of OER	*%age unit mapped with OER (approx)	Source URL
CSE322 Formal Languages and Automation Theory		Unit 1	FINITE AUTOMATA	Reading material (Pdf)	CSE322	80%	https://www.seas.upenn.edu/~cis2620/notes/cis262sl1-aut.pdf
		Unit 2	REGULAR EXPRESSIONS AND REGULAR SETS	Reading material (Pdf)	CSE322	80%	https://www.seas.upenn.edu/~cis2620/notes/cis262sl1-aut.pdf
		Unit 3	FORMAL LANGUAGES	--	CSE322	70%	https://www.cs.colostate.edu/~massey/Teaching/cs301/RestrictedAccess/Slides/301lecture05.pdf

OER(Open Education Resource)

Course Code	Course Title	Unit mapped	Broad topic	OER Type	Title of OER	*%age unit mapped with OER (approx)	Source URL
CSE322 Formal Languages and Automation Theory		Unit 4	CONTEXT-FREE LANGUAGES	--	CSE322	90%	https://www3.cs.stonybrook.edu/~pramod.ganapathi/doc/theory-of-computation/ContextFreeGrammars.pdf
		Unit 5	PUSHDOWN AUTOMATA AND PARSING	--	CSE322	90%	https://www3.cs.stonybrook.edu/~pramod.ganapathi/doc/theory-of-computation/ContextFreeGrammars.pdf
		Unit 6	TURING MACHINES AND COMPLEXITY	--	CSE322	70%	https://www3.cs.stonybrook.edu/~pramod.ganapathi/doc/theory-of-computation/TuringMachines.pdf

Main Perspective

The main perspectives are:

- Why are we learning Automata Theory?
- What would we do with it?

Why Study Automata Theory?

..contd..

Automata theory tells very important equivalence between

a language: some -- usually -- infinite set of strings

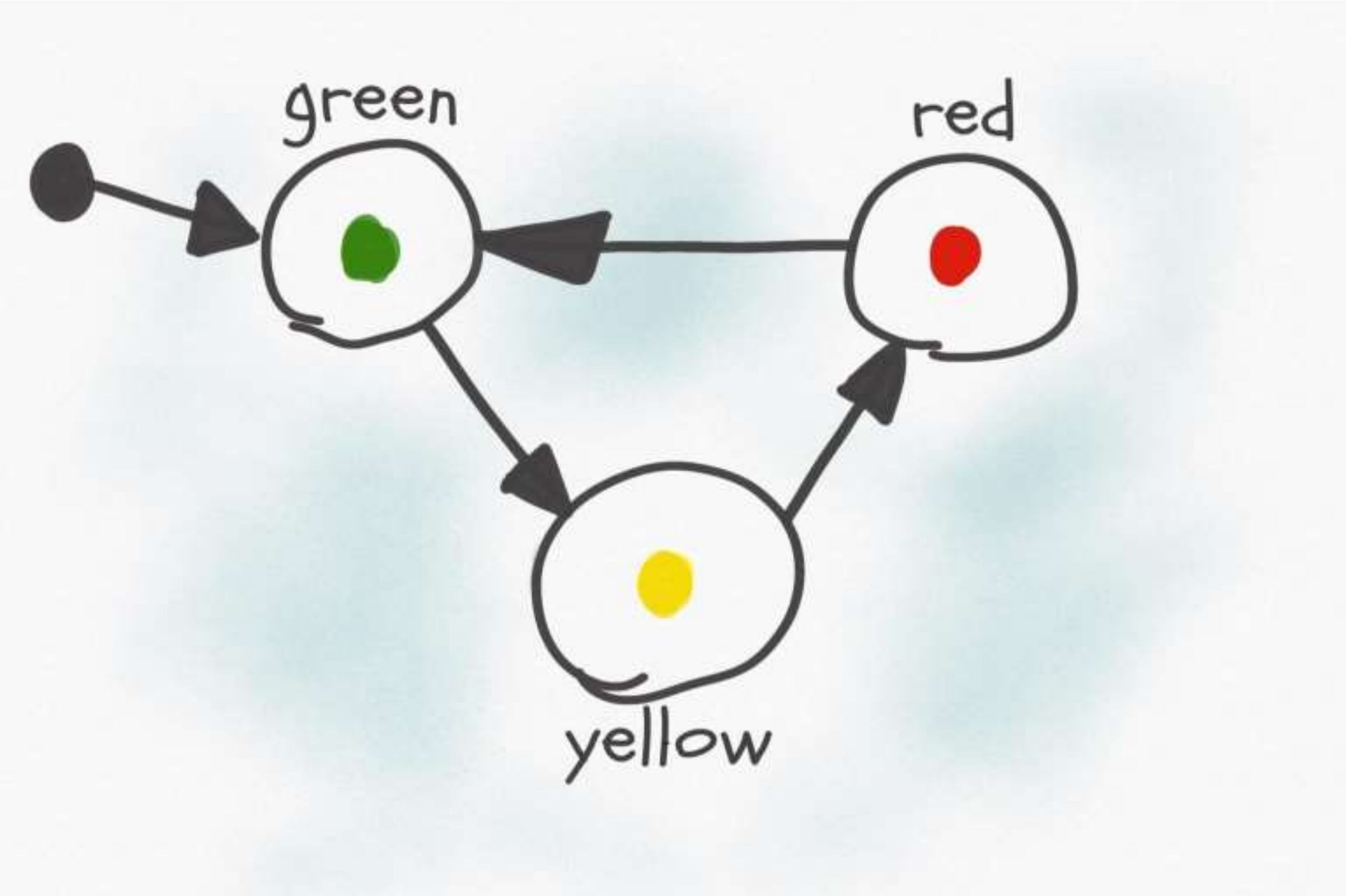
- a grammar: the finite set of rules to generate that language

- an automaton: the abstract processing device that can recognize that language

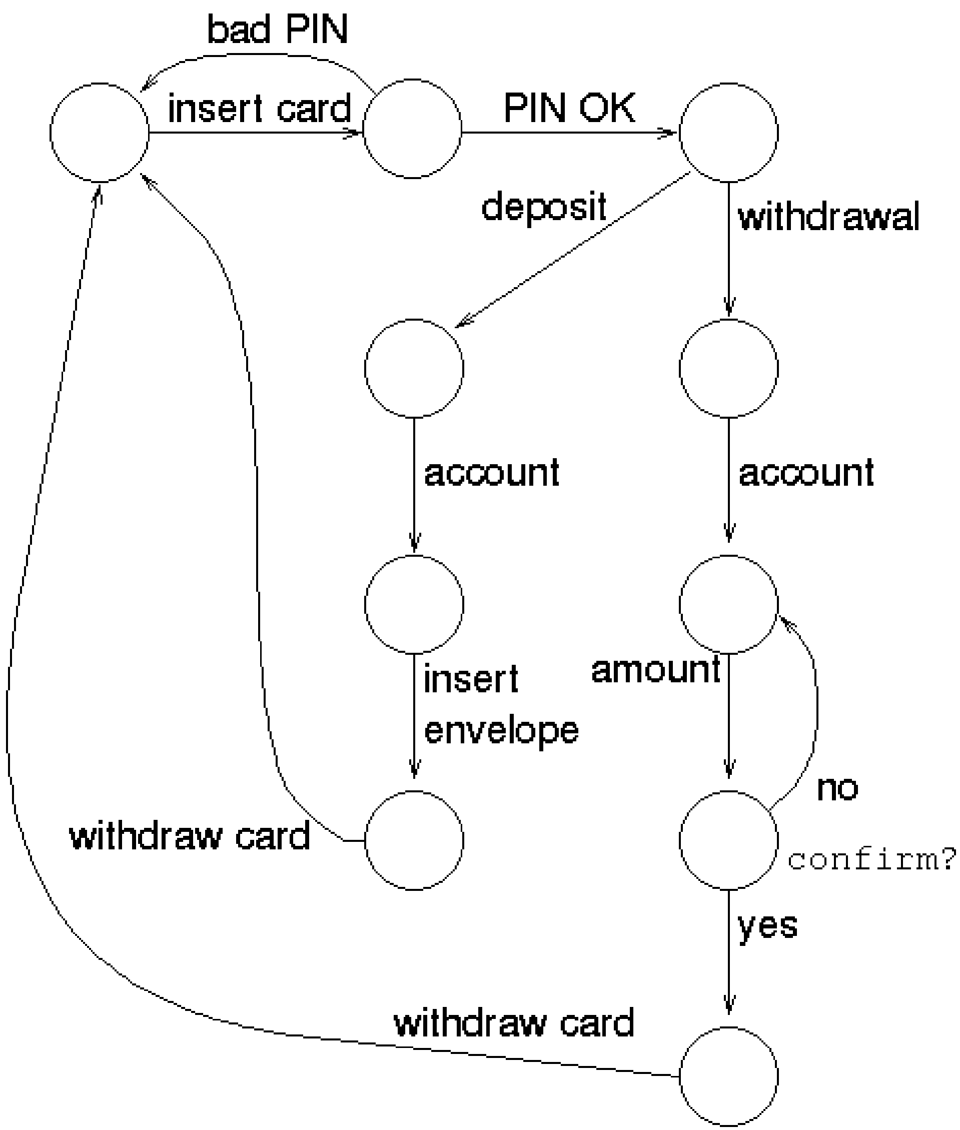
..contd..

- Automata theory is the study of abstract computational devices
- Abstract devices are (simplified) models of real computations
- Computations happen everywhere: On your laptop, on your cell phone, in nature, …
- Why do we need abstract models?

..contd..



..contd..



bad PIN
insert card
PIN OK
deposit
withdrawal
account
account
insert
envelope
amount
no
confirm?
withdraw card
yes
withdraw card

- Such devices are difficult to reason about, because they can be designed in an infinite number of ways

- By representing them as abstract computational devices, or automata, we will learn how to answer such questions

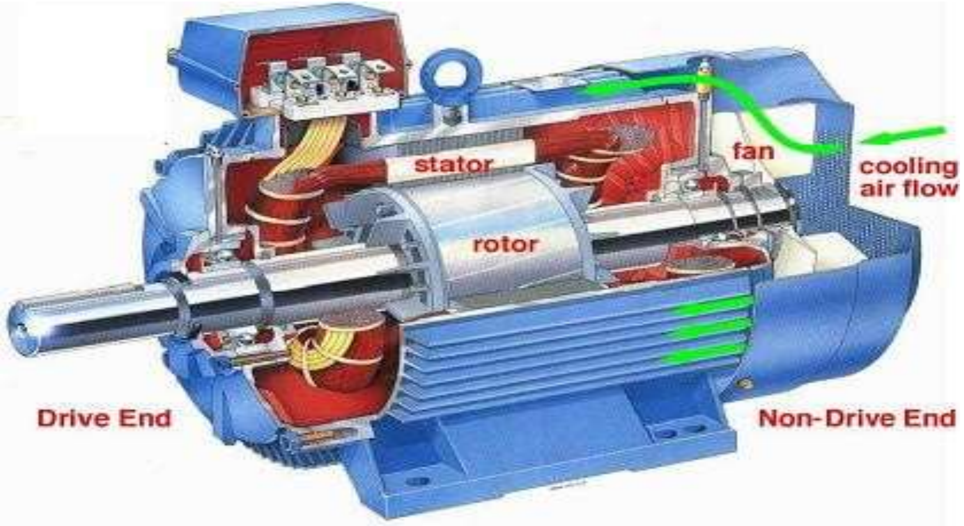
What would we do with it ?

There are numerous applications of Formal languages and Automata Theory like:

- Text processing, Compilers and Hardware Design
- Motors and Vending machines
- Sensors and Transducers
- Automata Simulators
- And many more ….

ATM MACHINE

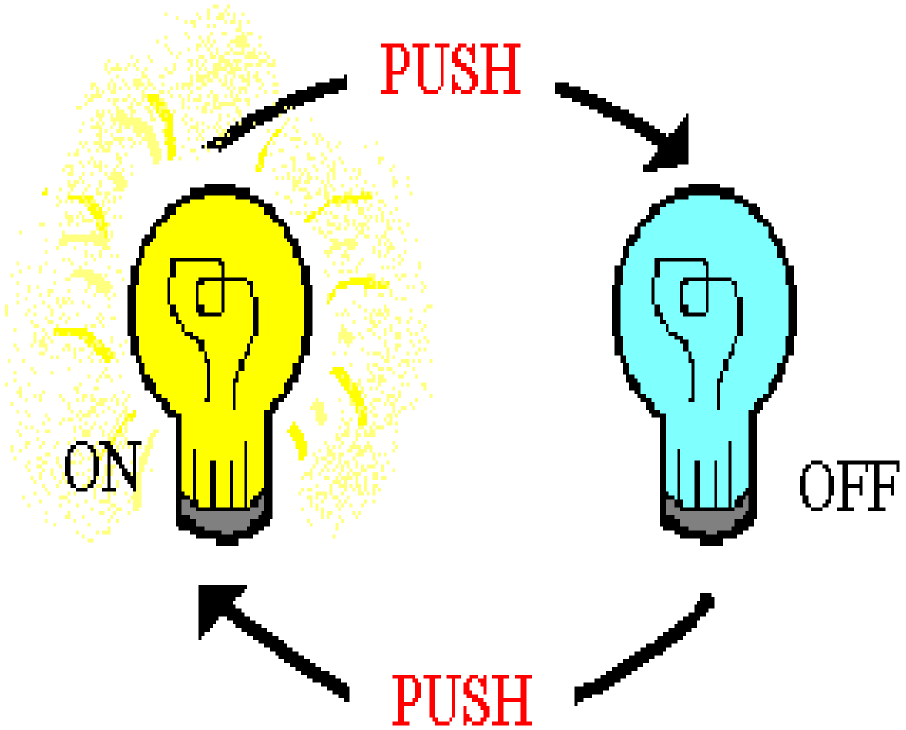
Motor



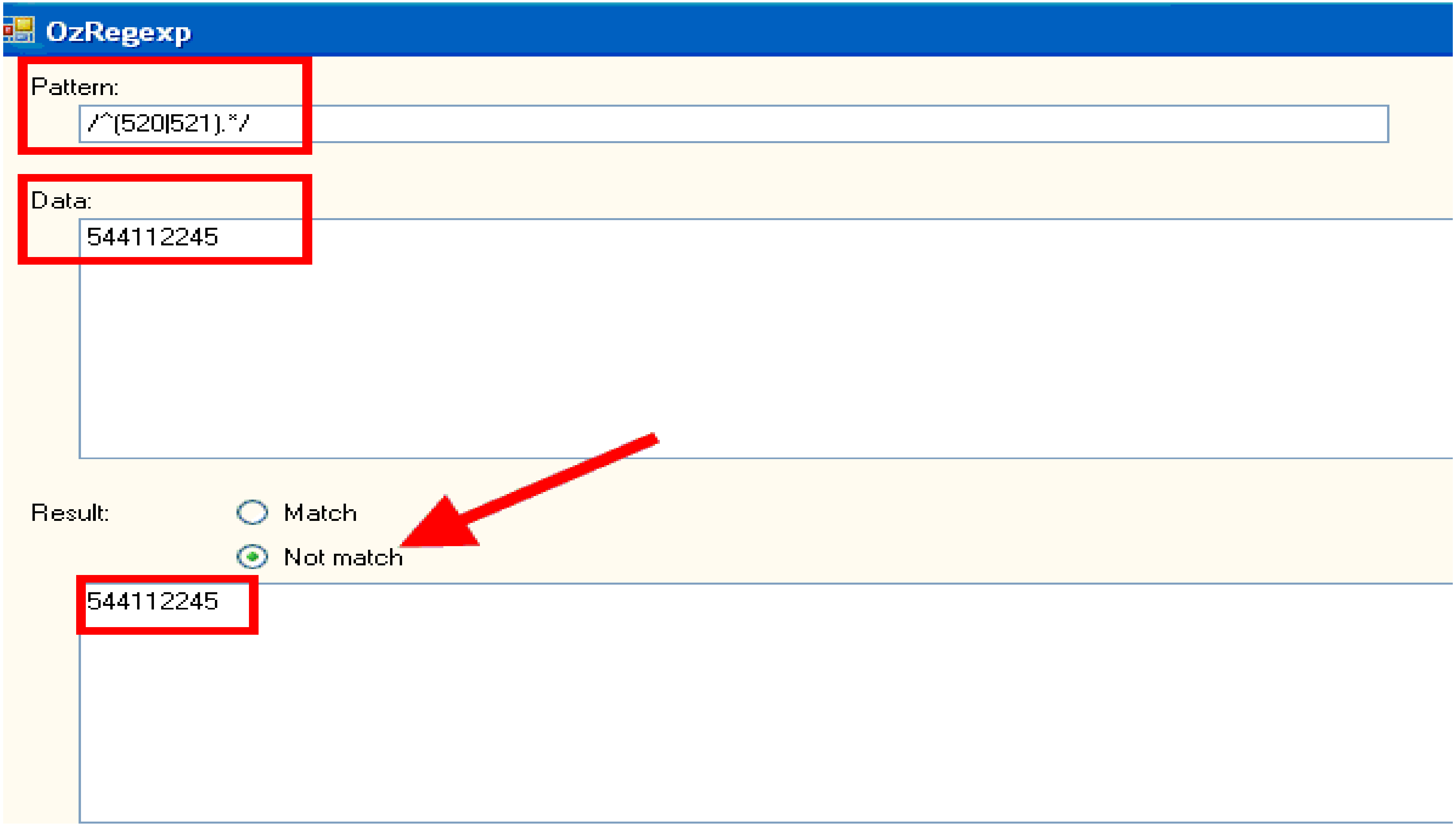
Vending machine

UNIT 1: Finite Automata

A lightswitch is an example of a finite automaton which has two states--on and off.



UNIT 2: Regular Expressions and Regular Sets



An **alphabet** is a set of symbols

{ a, b }

Sentences are strings of symbols

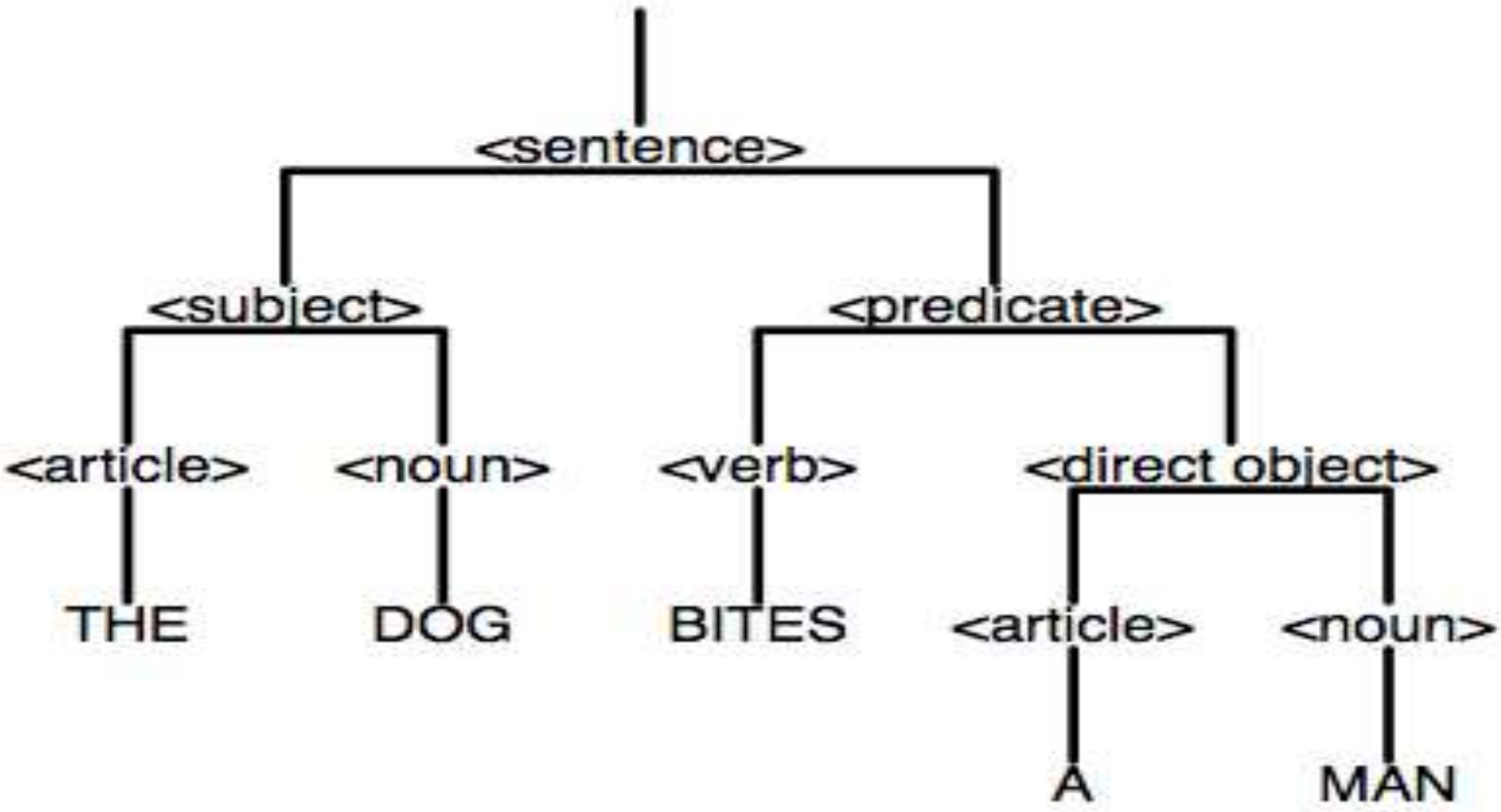
a, b, aa, ab, ba, a....

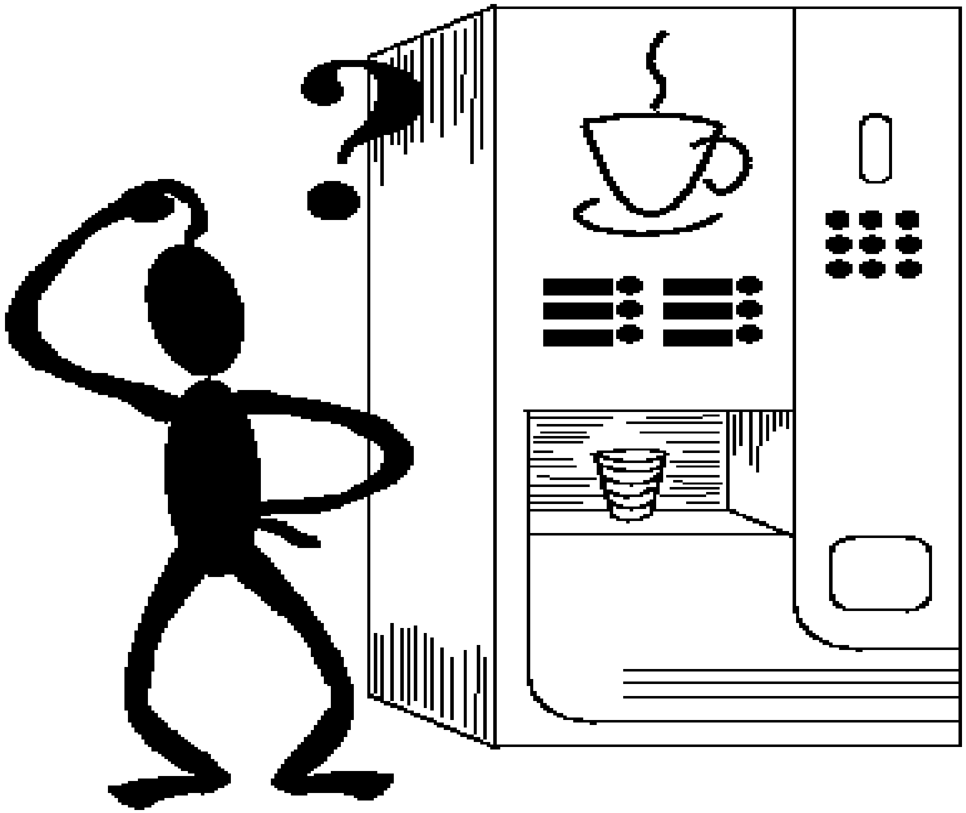
A **language** is a set of sentences

L = (aaa, abaa, aaba, bbb)

A **grammar** is a finite list of rules defining a language

$S \rightarrow aA$	$B \rightarrow bB$
$A \rightarrow bA$	$B \rightarrow aF$
$A \rightarrow aB$	$F \rightarrow \varepsilon$


<sentence>
<subject>
<predicate>
<article>
<noun>
<verb>
<direct object>
THE
DOG
BITES
<article>
<noun>
A
MAN



UNIT 6: Turing Machine and Complexity

Zero Lecture - Feedback

Follow this for getting instant solution for your Academic queries:

LPU_DIVISIONOFACADEMICAFFAIRS

Next Class : Finite Automata