Enhancing the learning ability using JFLAP for Theory of Computation Course

Pratibha S. Yalagi [1], Rashmi K. Dixit[2]

[1]Information Technology, Walchand Institute of Technology, Solapur, Maharashtra, India.
[2]Computer Scienece & Engineering, Walchand Institute of Technology, Solapur, Maharashtra, India.

[1]pratibhayalagi@gmail.com
[2]rashmirajivk@gmail.com

Abstract: The theory of Computation course is the core subject for Computer Science disciplines. It is based on the automata theory and design of machines like Finite Automata (FA) for regular languages, Pushdown Automata (PDA) for Context-free languages and Turing Machine for Recursively Enumerable languages. Most of the part includes theoretical concepts like theorems, proofs and examples. The most promising part of this course is to solve examples according the given theory concept for a particular machine design. Teaching and learning this course is based on the imagination of theoretical concepts with the design of machine part. For a novice computer science student it is difficult to connect with the theory and a design for solving examples. The study proposes an effective way of teaching this course using Java Formal Languages and Automata Package (JFLAP) tool. It is a preexisting software tool created by Professor Susan Rodger and her students at Duke University. It allows designing and running machines like finite automata, pushdown automata and turing machine with the theorem proofs. It also focuses on conversions of languages from one form to another form easily.The experiment is conducted for the second year computer science and engineering students for the course of Theory of Computation (TOC). The results show that, the ability of problem solving among students increased effectively with more understanding level. The effectiveness of the experiment is verified by the feedback from students and the university results. It is observed that the learners can evaluate them selves for the problems using JFLAP.

Keywords Theory of Computation, Automata Theory, Simulation, FA Design and evaluation, Visualization.

1. Introduction

Usually in computer science education learners gets only superficial knowledge by the theoretical concepts. Especially the subjects related to formal languages and automata theory which plays major role in the construction of compilers and designing programming languages. The course theory of computation is also focuses on formal languages and automata theory and students cannot easily visualize theoretical constructs by the traditional teaching. In traditional teaching problem solving is done by pencil and paper method where the students do not understand actual concept through visualization. They cannot easily evaluate their solved problems to determine if their solutions are correct. Hence they depend on others for the solutions especially on teachers. [12] This course contains more mathematics and also requires discrete mathematics concepts as prerequisite. So the weaker students need to work on more examples and problem solving to acquire subject knowledge. The automata theory uses more mathematical notations to represent languages, theorems and proofs.

The software tools like JFLAP are more useful for students to visualize theoretical concepts and also to understand annotations and terminology behind formal languages and automata theory. The JFLAP tool provides an interactive visualization for automaton constructs and gives feedback for the construction of automaton. It works as a simulator for automata constructs like Finite Automata for regular languages, Pushdown automata for context free languages and also for Turing machines. While solving the problems the students receive the feedback and become more familiar with the mathematical notations and formal representations easily with the proper visualizations.[13]

The experiment is conducted for second year Computer Science and Engineering undergraduate students for the course of theory of computation. The concepts are taught in theory class and the problems are solved in tutorial sessions with a small batch of students. During tutorial sessions JFLAP is used for simulating the automaton and to receive feedback for constructed machine. It is observed that the students involvement increases in solving the examples as they can easily visualize the theoretical notations and concepts with proper simulations.

2. Literature review

Most of the computer aided learning tools are available for teaching-learning process. Many tools are available for mathematics, discrete mathematics and computation theory. One of them mostly used tool is Mathematica [3]. which is focused in education and research both. It offers most of the numerical and graphical libraries and highlevel interpreted programming language. The tool also contains Combinatorica [4], offering a collection of over 450 algorithms for grapth theory and discrete mathematics. Simpler, non-commercial tools are also available on the Web. For example, animated flow of absic algorithms with animationsof sorting algorithms as well as the source code used can be found on the websites [5]. For the subjects like computer networks, animations of the various data-link layer protocols can be found at [6]. Several notable

tools exist in the field of theory of computation for regular languages and automata theory, one of the tools is FIRE engine [7]. It offers C++ class library for implementing regular expression and finite automata algorithms. Grail+ [8] is also one of the symbolic computation environments for regular expressions, finite-state machines, and other formal language theory objects. Grail+ can be used for inputting machines or expressions, convert them from one form to another. It can also be used for minimization, complementing, and for making them deterministic. Tools like Automata Simulator [10] and SoftLab [11] are used in the field of fundamentals of automata theory. Automata Simulator provides interactive generation and simulation of finite state machines. This simulator supports all types of finite state machines: automata with binary output (DFA, NFA, and NFA-null transitions) and automata with general output. Softlab is a fully distributed e-learning tool and is an extension of Automata Simulator. It provides better interaction between the student and the teacher as a supervisor to monitor the student's progress and even assist in modeling more complex examples. JFLAP [9] is preexisting software tool for theory of computation, experimenting with formal language topics including Finite Automata, Pushdown Automata, Mealy Machine, Moore Machine, Turing Machine, Multi-Tape Turing Machine, Grammar, L-System, Regular Expression, Regular Pumping Lemma and Context-Free Pumping Lemma.

3. Java Formal Languages and Automata Package (JFLAP)

JFLAP is a software tool used for teaching formal languages and automata theory. JFLAP is created by Professor Susan Rodger and her students at Duke University. It provides interactive visualizations for the construction of finite automaton, pushdown automaton and turing machine. It guides students to visualize the difficult mathematical notations and abstract notions of theory of computation. Using JFLAP complicated examples can be solved step wise easily. Use of JFLAP saves time by providing feedback of solved examples. Students can use this to test their constructed automatons for correctness and it also increases accuracy.

It deals with finite automaton, mealy machine, moore machine, pushdown automata, turing machine, grammar, regular expressions and pumping lemma.

Features of JFLAP

- Allows defining a grammar and checks whether the given string is a member of that grammar or not.
- Allows constructing DFA and testing one or more strings for acceptance or rejection.
- Provides step-by-step processing of string in to an FA according to the FA function
- The NFAs can be converted in to DFA
- The FAS can be minimized
- Allows combining automata using union and concatenations
- It helps to visualize the nondeterminism concept by keeping all its possible configurations
- It also allows disabling some of the paths in NFA for successful processing of the provided string.
- It also allows converting a right linear grammar in to NFA with step wise construction
- Pumping lemma is made easy by the gaming visuals
- Easily determines membership of a string in a language defined by a CFG using brute-force parser and shows a parse tree with left-most derivation.
- Can construct PDA and string membership can be tested easily with step-wise confirmation
- It also allows constructing Turing machine with string membership testing

The proper use of JFLAP and instructions are provided in the manual "JFLAP activities for Formal Languages and Automata" by Peter Linz and Susan Rodger. The manual comprises of all the instructions and details about the theoretical concepts of automata theory and their implementation in JFLAP software. [14]

Uses of JFLAP made easy to study and to teach theory of computation course. It saves time and provides accuracy. As it provides feedback and response immediately students can immediately verify their solved problems and corrects them.

4. Methodology

The course Theory of Computation is fully based on automata theory based on finite automaton, pushdown automata and Turing machines. It focuses on the base of Discrete Mathematics notations and complex automata theory. In traditional teaching method theory is taught in the class and examples are solved by the students in tutorial sessions under the guidance of a teacher. The learners usually face problems while solving exercise problems where they are unable to correlate theoretical part with the problems. Some good students solve such problems and others follow them, hence it loses concentration of students and the interest in the subject reduces. Our experiment focuses on enhancing learning abilities of students through the effective use of visuals of theoretical concepts and providing the interaction during problem solving. The JFLAP tool provides visualizations of constructions of automatons like Finite automata, Pushdown automata and Turing machines. It also provides interaction with stepwise process and feedback about the membership of a given string immediately where students can verify their solved problem and correct it immediately. It reduces the time of problem solving and enhances the learning ability of a novice learner.

This experiment is conducted for second year undergraduate Computer Science and Engineering students. The whole class is involved in this experiment (around 70 students). According to the prescribed curriculum by

university weekly three theory classes and one tutorial for problem solving is mentioned. In theory class theoretical concepts are explained by the teacher with few examples and problems. A batch of 15 to 18 students is there in tutorial class where they are solving problems based on the topics covered in theory class. As the strength of the batch is very less as compared to theory class, teacher can guide students properly for solving problems. It has been observed in the previous year students that they are not able to solve problems efficiently by their own. Some good students are solving problems and others are following them. Due to this students concentration and interest in the subject is reduced which effects directly on their results. Therefore to improve the learning ability of students with the interest JFLAP tool is used during tutorial sessions in the lab. The learners are using this tool for solving complex problems and getting stepwise progress with transition diagrams and detailed acceptance or rejection of a string according to the state wise update of a designed machine. The tool guides students in understanding the concepts of mathematical problems of TOC using proper visualization of construction steps. Its interactivity improves student's interest in problem solving through immediate response. The verification and correction can be done immediately during the tutorials improves the speed of problem solving and automatically more problems are solved during the session, which directly increases concentration and creates interest in the subject.

5. Implementation

The JFLAP tool is used in tutorial sessions for solving problems and for getting responses immediately. The students use this tool for stepwise process of construction of Finite automata with proper input and verify the given input string for acceptance or rejection. JFLAP supports construction of Finite automata, Pushdown automata and Turing machine. It also allows grammar derivation and also finds a regular expression. It converts Nondeterministic FA to Deterministic FA and Regular expression to Nondeterministic FA and also minimizes the FA.

The theoretical concepts are taught in theory class in a traditional way. The theory of computation with examples is explained in front of the class with more than 70 students. The examples are solved in tutorial sessions with a small batch of around 15 to 18 students. In usual tutorial sessions exercises are solved by students under the guidance of a teacher. But as a teacher cannot concentrate on every student personally, students solve by their own with their knowledge. But it becomes difficult for them to check their solved problems for correctness. Usually they check it by comparing other students' answers or by asking to the teacher. It is very time consuming and also students will lose their concentration and it automatically decreases the students' interest in the subject. Using JFLAP students can easily verify their answers and correct them immediately. Their doubts can be clarified with stepwise process where each state and next state with clarification is represented properly by JFLAP. The stepwise process clarifies most of the concepts. It becomes easier for the learners to go ahead with proper guidance by the tool.

It is observed by the results of the university exam that the students who have used JFLAP scored more than the students who have been taught in a traditional way. Due to the proper use of a good visualization and interactive problem solving techniques the results are improved.

Figure 1 shows the JFLAP main menu. Mainly the options Finite automaton, Pushdown automata, Turing machine, Grammar, Regular expressions, pumping lemma are used in our course of Theory of Computation.

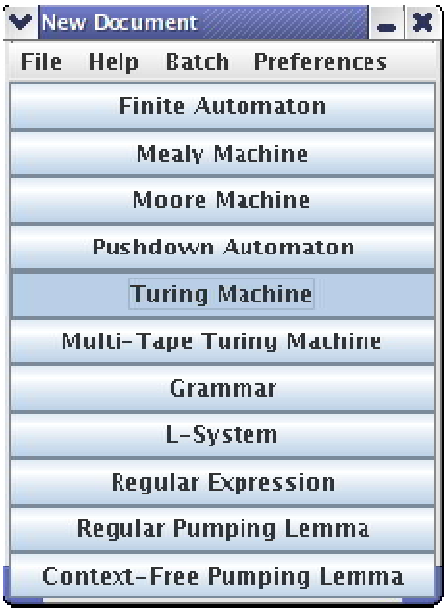

Fig. 1 JFLAP Main menu

JFLAP allows testing of a membership of a string in multiple runs where more than one string can be tested at a time. The Figure 2 and 3 are showing two different examples solved with the input strings for determining acceptance and rejection by Finite automaton.

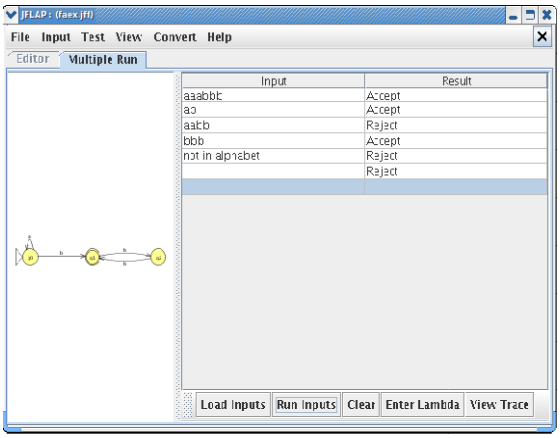

Fig. 2 Example string testing for membership

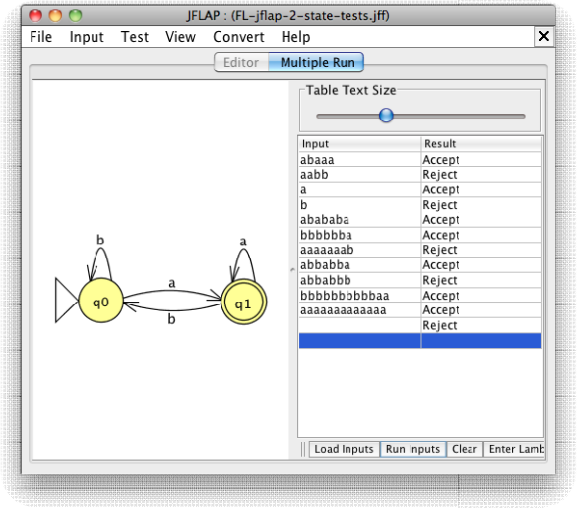

Fig. 3 Example string testing for membership for a string ending with a

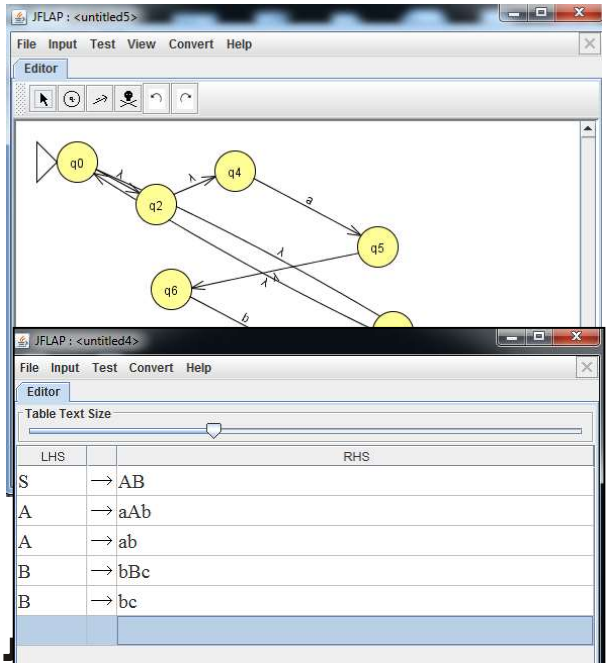

Fig. 4 Regular Expression to NFA Conversion

It is very useful for constructing the Nondeterministic FA for a given regular expression where Kleen's theorem is applied with concatenation, union and closure operations are used. Figure 4 shows example of Regular expression to Nondeterministic FA. Another example is provided for grammar representation in Figure 5.

Fig. 5 Grammar representation

6. Results

The results are observed by student's university results. Two years results are compared for this experiment. Academic year 2014-15 and academic year 2015-16 are compared and it is observed that students in 2015-16 have scored more than the previous year students. The 2014-15 year students have undergone through traditional teaching and learning methodology and for students of academic year 2015-16 JFLAP tool is used for problem solving in tutorial sessions. A remarkable improvement among the students is observed.

The following Table 1 shows results observed in year 2014-15 and 2015-16. The analysis is done based on their marks range for Theory of Computation course.

Table 1. Result analysis

Range	No. of Students 2015-2016	No. of Students 2014-2015
Below <= 40	3	6
41-50	8	20
51-60	20	17
61-70	20	19
71-80	17	11
81-90	5	1
91-100	3	0

It has been observed that the marks range is increased by around 20% to 30% with the use of this tool. The following Figure 6 shows graphical representation of result analysis. This analysis depicts that the number of students scoring more than 70 is increased drastically. Few students have scored more than 90 marks.

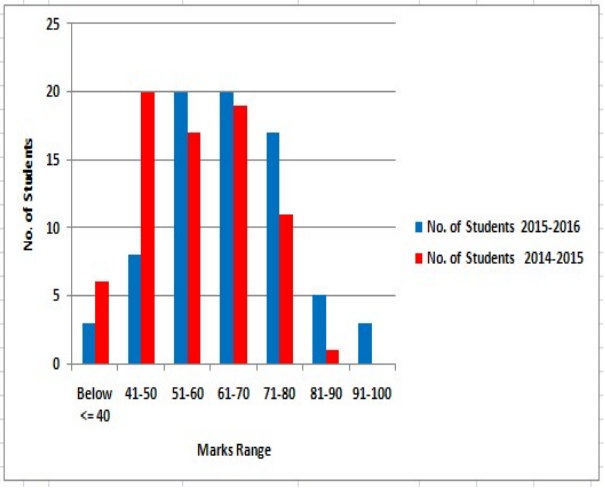

Fig. 6 Graphical representation of the result analysis

Student's feedback is also collected where students agreed that their concentration while studying this subject and interest in the subject improves. Overall result is improved by the use of JFLAP tool. It also enhances learning ability by improving problem solvability and understability among the students.

In order to collect students' feedback for the effect of use of JFLAP tool, a survey questionnaire was designed with questions based on efficiency of the tool and its ease of use. Students acknowledged with the answers of the questions. This was a five point likert scale type with ten questions. The scales varied from strongly agree to strongly disagree. Questions were designed to know whether the tool is really helped students, to improve their problem solving ability.

7. Conclusion

Through this experiment it is observed that, for the theoretical course like Theory of Computation where mathematical notations are used can be taught easily using visualization technique. The problem solving ability is improved through its immediate response where the learners can easily verify their answers. It decreases the dependability of student while studying. JFLAP tool is the useful tool for construction of automata theory and for checking membership of a string in the language. It is also helping the learners in conversion of Regular expression to Nondeterministic FA and for grammar derivation too. It has been observed that the results improved by more than 20% in the final exams too.

Acknowledgment

This paper is made possible through the help and support of Management, Principal, and Head of department, PLC coordinator, family and friends'. The product of paper is not possible without them.

References

[1] Ivan Budiselic, Sinisa Srbljic and Miroslav Popovic, "RegExpert: A Tool for Visualization of Regular Expressions", IEEE International Conference on "Computer as a Tool", EUROCON 2007.

[2] S. Srbljic, Compiler Design 1: Introduction to Theory of Formal Languages, Automata, and Grammars, (original title in Croatian: Jezicni procesori 1: uvod u teoriju formalnih jezika, automata igramatika"), 2nd Edition, Element, Zagreb, 2002, pp. 46-50.

[3] S. Wolfram, the Mathematica Book, 5th Edition, Wolfram Media, 2003.

[4] Pemmaraju and S. Skiena, Computational DiscreteMathematics: Combinatorics and Graph Theory with Mathematica, Cambridge University Press, 2003.

[5] B. aneva and D. Thiébaut, "Sorting algorithms," Smith College, http://maven.smith.edu/~thiebaut/java/sort/demo.html.

[6] A. Jacobsen, "Data-link network protocol simulation," University of Birmingham, School of Computer Science, http://www.cs.bham.ac.uk/~gkt/Teaching/SEM335/dlsim/Simulation.html.

[7] B. W. Watson, "The design and implementation of the FIRE engine: a C++ toolkit for finite automata and regular expressions," Computing Science Note 94/22, Eindhoven University of Technology, Netherlands, 1994.

[8] Grail+ Project homepage, http://www.csd.uwo.ca/Research/grail/index.html.

[9] S. H. Rodger and T. W. Finley, JFLAP: An Interactive Formal Languages and Automata Package, Jones & Bartlett Publishers, Sudbury, MA, 2006.

[10] Skuliber, S. Srbljic, and I. Crkvenac, "Using interactivity in computer-facilitated learning for efficient comprehension of mathematical abstractions," Proceedings of the EUROCON 2001, International Conference on Trends in Communication, Bratislava, Slovak Republic, July, 2001, vol. 2/2, pp. 278-281.

[11] Skuliber, S. Srbljic, and A. Milanovic, "Extending the textbook: a distributed tool for learning automata theory fundamentals," Proceedings of the 9th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2002), Dubrovnik, Croatia, September, 2002.

[12] Susan H. Rodger, "JFLAP-Software for Experimenting with Formal Languages" Department of Computer Science, Duke University, July 11, 2007, 2007 Premier Award

[13] Eric Gramond and Susan H. Rodger, "Using JFLAP to Interact with Theorems in Automata Theory" Duke University, Durham, CISE-9634475.

[14] Peter Linz and Susan Rodger, "JFLAP activities for Formal Languages and Automata" manual, Duke University, Februaury 2011.