Problems In Understanding Abstract Concepts Related To Regular And Context-Free Languages

ZEESHAN ALI RANA, MUHAMMAD ASHRAF IQBAL, YASSER HASHMI
Lahore University of Management Sciences (LUMS)
Opposite Sector U, DHA, Lahore
PAKISTAN

Abstract: - Understanding abstract models in automata theory is important in the fields of computer science and digital circuit design. In automata theory, problems are categorized in different classes of computability, from simpler to harder. Problems which lie in class of regular languages are simplest kind of problems. Problems which lie in class of context-free languages are relatively harder. Students tend to confuse the relationship between these two classes of languages and between their members. Questions related to the closure properties of regular and/or context free languages pose particular problems. Students make mistakes performing operations like intersection, union and concatenation on sets of regular and context-free languages. For this research the students of automata are asked to solve some examination and assignment problems related to closure properties of the above said classes. A questionnaire, seeking descriptive answers, is used to study 1) the mental process 2) the structure of mental concept map and 3) line of reasoning followed while attempting a problem. This study reveals that the students make mistakes because their abstract concepts are not correctly structured and in particular do not support logical inferences. Students also have a tendency to apply the concepts mechanically instead of applying them meaningfully. Finally, students fail to actuate proper conceptual link at proper time.

Key-Words: - Abstract Concepts, Motivation, Line of Reasoning, Regular Languages, Context-free Languages, Think Aloud.

1. Introduction

Automata and computability theory have a major role in understanding of computer science concepts. Circuit designing also needs knowledge of automata concepts [1]. Automata theory categorizes problems in sets of simpler problems and harder problems. Simpler problems can be solved using machines which do not have any temporary memory. Relatively harder problems can not be solved without memory but can be solved using the machines with stack memory. Such machines can be called stack machines. The set of simplest problems is called set of regular languages (RLs) whereas the set of relatively harder problems is called set of context-free languages (CFLs). There are even harder problems than the context-free languages which can not be solved using stack machines. This paper only focuses the regular and context-free languages and presents the problems noticed during the learning of concepts related to the above two sets of languages only. Specifically, the paper portrays the problems in understanding the set operations like union, intersection, complement and concatenation on the sets of regular and context-free languages.

The concepts related to computability theory are considered abstract and cognitively complex [2, 3]. In this paper we will follow a definition of concepts suggested by Novak [13]. A concept consists of a name or a tag used to access the concept. Stored attached to the tag are attributes or properties of the concept. It is these properties that support any logical inferences linking a concept with other concepts. In this sense an abstract concept is a concept whose attributes are not based on specific instances but rather are based on classes of objects. In other words an abstract concept has properties that refer to other concepts rather than concrete instances. We regard these abstract concepts as particularly challenging to learn and to reason about. Thus these concepts are congnitively complex because people fail to handle mental manipulations needed [3]. Studies reveal that learning abstract concepts is a difficult task [2, 4] and the cognitive complexity is one of the factors causing frustration and lack of interest to learn [2]. In addition, learning theories affirm that students construct knowledge actively and it can not be absorbed passively using text books and lectures [6, 7]. So the caused frustration restrains the students to actively construct the knowledge and perform meaningful learning.

Acknowledging the complexity of the

concepts, students still need to motivate themselves in order to cope with the frustration and continue learning meaningfully. In order to keep themselves motivated, students set themselves some goals which could be achievement goals, learning goals and performance goals [5]. Goals are set according to the student's approach towards learning. This approach is another factor which affects meaningful learning. Chin [8] discusses two approaches towards learning; intrinsic and extrinsic. Intrinsic approach is learning due to a desire and extrinsic approach is learning for the sake of others' wish, the others can be parents or teachers [8]. Students with intrinsic interest seem to have learning goals in their minds and have interest in studies, and hence are better learners. Study by Hazzan [3] exhibits that even such good students make mistakes when asked to solve problems related to abstract concepts. Hazzan notes a strategy where students reasonning with abstract concepts prefer to convert them to concrete examples. After reasoning with concrete examples they translate the inferences reached back to the abstract domain. In other words they reason by example. We note a similar strategy in our students reasoning about automata theory. In addition, Chesnevar et al. [2] beleive that students fail to understand concepts particularly when reasoning about them in real time because their focus is on the mechanical application of a rote learnt procedure rather than a meaningful understanding of the concepts.

The basic aim of this study is to find where things go wrong in knowledge structure of students. For this reason we collected a set of exam questions which target a specific set of links in the student's concept map. Automata theory is taught as a core course to computer science majors in most of the universities. These concepts are used to build further knowledge of computer science. So meaningful learning of these concepts is critical in understanding of computer science concepts. Text books of Automata theory typically keep their narration at the abstract level without linking these abstract concepts to their concrete instances. We suspect that this may contribute to a rote learning of absract concepts so that they do not support error free logical inferecing nor are students neccessarily aware of how reasoning must change depending on the level of abstraction. We further suspect that sticking to an abstract narration also interferes with the subjective level of interest of the student. This study reveals that the abstract narrative of the text books leaves some students behind. Specifically those whose reasoning fluctuates between abstract concepts and concrete examples and who fail to discover themselves the contrast between these two.

The paper is organized in the following manner. Section 2 gives a brief background needed to understand the concepts used in the paper. Section 3 discusses the way this research was designed and conducted. Section 4 discusses the results of the study and the conclusion is presented in Section 5

2. Background

In automata theory problems are termed as languages and they are categorized into different classes with respect to their difficulty level. Formally, a language is a set of strings, for example "set of all the numbers ending with 0" can be considered one language and a string is a finite sequence of characters.

Regular languages are simplest kind of problems. Problems in the class of regular languages can be solved using finite state machines having no temporary memory. We call such machines Deterministic Finite Automata (DFA) or Non-Deterministic Finite Automata (NFA) in our concept map in figure 1. For example, counting the number of instances of a character in a particular string, determining whether a number is even or odd are members of this class.

Class of context-free languages includes all the members of regular languages as well as some relatively harder problems. The more sophisticated kind of counting will fall into this category. Like determining whether every open brace has a corresponding closing brace in the program or determining whether a string is reverse of itself are kind of problems this class contains. Solution to such problems needs some temporary memory to store information. These solutions can be obtained by using finite state machines having stack memory only. We call such machines as Push-Down Automata (PDA) in our concept map in figure 1.

Closure properties of a class show which operations on members of a class will yield the member of the same class. If an operation on any two members of a set always yields an element of the same set, then that set is said to be closed under the performed operation [9, 10]. For example the set of natural numbers is closed under addition but is not closed under division [10]. Similarly the sets of regular languages and context-free languages have some closure properties as well. Regular languages are closed under union, intersection, complement, concatenation and some other operations. In this paper we will focus on the above listed four

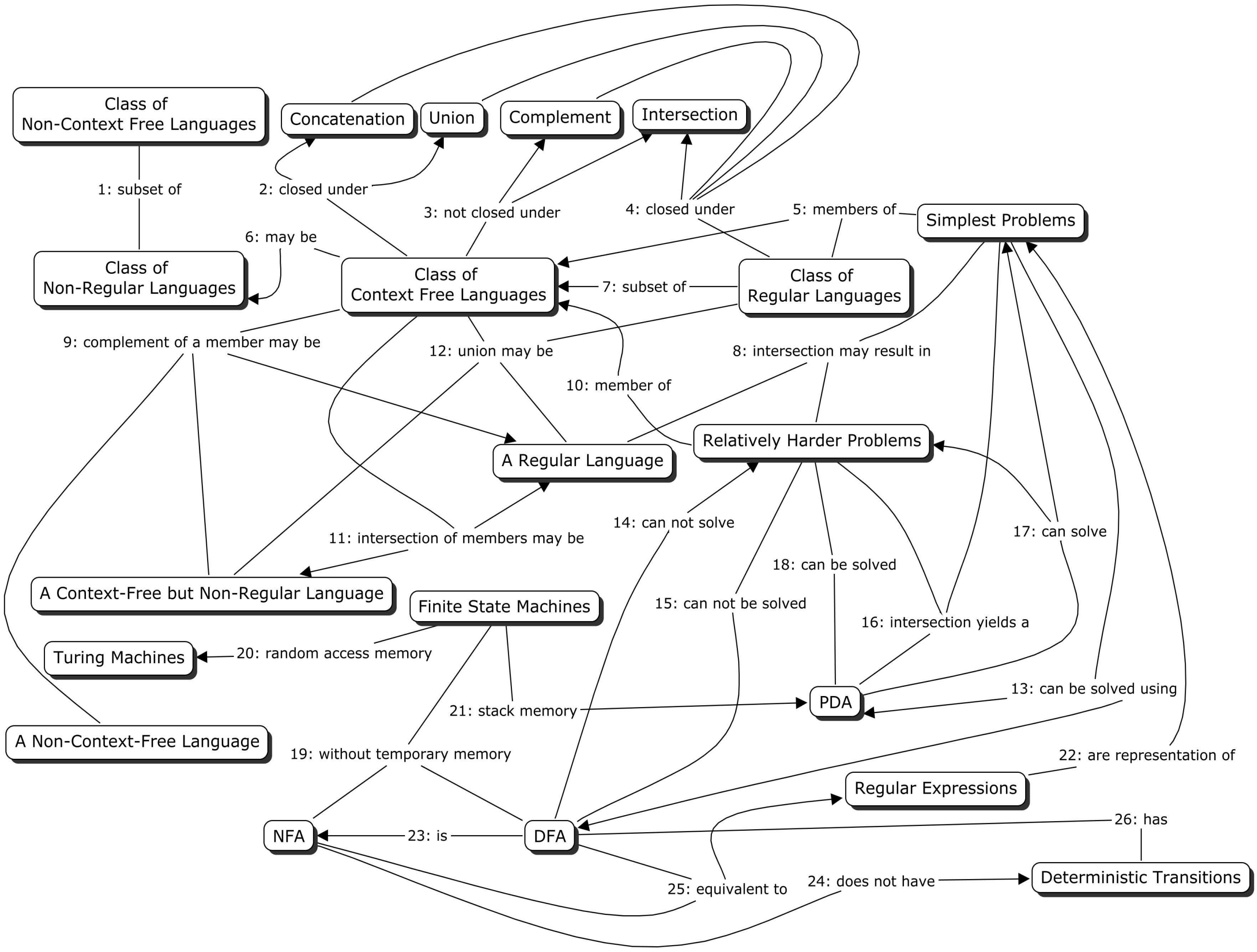

Fig. 1 Concept Map Showing the Relationship of Classes of Languages and the Tools Used to Solve Them

operations only. Context-free languages are closed under union and concatenation but complement of a context- free language may fall outside the class of context-free languages. Similarly intersection of two members from the class of context-free languages may not yield a member from the same class.

The concept map regarding relationship of the languages and the tools needed to solve the problems of each class is shown in figure 1. It shows how the automata theory concepts are linked together and the structure expected in a student's mind. The links are numbered in the figure just to refer them in coming sections.

3. Research Design

We collected data of 30 students of the course Automata and Complexity Theory taught at Lahore University of Management Sciences, Lahore, Pakistan in academic year 2006-07. The class was a hybrid of graduate and 4th year undergraduate students. We made no distinction between these groups and pooled the results.

3.1 Research Hypotheses

Our research hypotheses are:

1 For some questions ability to generate the right concrete example will lead to a succesful translation of inferences at the abstract level and the correct answer. Here ability to generate an example will predict a correct answer.
2 In some questions reasoning using concrete examples is more likely to lead to an incorrect inference at the abstract level. Here using examples will hinder reaching a correct answer.
3 Lack of motivation is a hindrance in learning of these concepts.

3.2 Examination, Assignment and Think Aloud Interviews Questions

We collected 10 questions from assignment and mid term of Automata offered at LUMS, Lahore, Pakistan. These questions were related to closure properties and relationship of classes of languages.

Table1 Questions Asked In Examination and Assignment

Q. #	Questions
	State whether True or False. Justify your answer to get any credit.
Q. 1	Complement of a CFL may be non-CFL
Q. 2	Intersection of a regular language and CFL is always a regular language
Q. 3	Concatenation of a regular language and a CFL is always CFL, but not regular
Q. 4	L4 = L1∩L2∩L3, where L1 and L2 are regular and L3 is CFL. It is possible that L4 will be a RL
Q. 5	Let L4 = L1 L2 L3. If L1 and L2 are regular and L3 is not regular, it is possible that L4 is regular
Q. 6	L2 = Complement of L1, where L1 is a CFL. It is possible that L2 will be a regular language
Q. 7	L4 = L1UL2UL3, where L1, L2, L3 are regular languages. L4 will always be a context free language
Q. 8	Intersection of two non-regular languages is always non-regular
Q. 9	Intersection of two CFL's is CFL
Q. 10	Subset of a CFL is always a CFL

Table 2 Questions Asked In Think Aloud Interviews

Q. #	Questions
Q. 1	A coin counter which counts 6 coins of value 5. Is it a regular language (RL) solver?
Q. 2	A soda 'can' releaser which releases a 'can' if input is 1. Does this releaser solve a regular language?
Q. 3	Make a dispenser which takes coins of value 5 as input, counts if there are 6 coins entered and if yes then release a 'can'. Use the above two machines to make the dispenser. Which operation do you need to perform?
Q. 4	Is the problem solved in Q. 3 a regular language? Take its union with any CFL. Will the results be regular ever? Which example did you think while taking union and why?
Q. 5	A robot which identifies red balls and separate outs them from balls of all colors. Is the problem solved by the robot a regular language?
Q. 6	If the robot needs to separate out as many red balls as blue balls, will it be a regular language?
Q. 7	The above told dispenser now asks even number of questions before asking you for money. If you answer more than 50% questions correctly, then it releases a soda can without asking for money. Is it a RL? Or CFL but not RL?
Q. 8	If you answer 50% of the questions correctly then it asks you to enter 10 rupees and on entering coins which sum up to 10 rupees, it releases a soda 'can'. Was there any additional machine in the dispenser of Q. 3? Which operation was performed if some additional machine has been added?
Q. 9	If you answer less than 50% correctly then it asks for full payment to release a soda 'can'. Which operation you perform this time if some additional machine is added?
Q. 10	Are the machines in Q. 8 and Q. 9 collectively the complement of the machine in Q. 7? Will the complement be a CFL?

Table 1 lists the questions asked in the exam and the assignment. Q1 in table 1 can be solved without using an example and can even be rote learnt since it is a property of the class of CFLs. Q2 targets the links 5, 7, 8, 13 and 16 of figure 1. It is the property of languages that intersection of a DFA and a PDA will always be a PDA and a PDA can solve the simplest problems as well. An example is not needed to solve the problem. Q3 and Q4 target the links 5 and 7 whether the students use this link at appropriate time. This question again does not need use of example but if a student picks a language which belongs to class of regular languages, he / she can solve the problem. Furthermore, Q4 attempts to check whether students know if regular languages are closed under intersection or not. For Q5, students need to conceptualize the resulting concatenated language which they should find difficult if they do not use an example. The reason of the difficulty is that here are many abstract concepts to cater and making inference between them at abstract level will be a difficult task. Q6 checks the students' knowledge on the property of regular language that they are closed under complement. Q6 can easily be solved without using examples. Q7 simply checks if students can identify the relationship between the classes of CFLs and RLs, in addition it checks closure property of RLs. This question is a simple question and if the students have the links 4, 5 and 7 in their concept map they can solve the problem without any example. Q8 asks students to think abstractly about the class of regular and non-regular languages. We expect that students using the examples will be comfortable solving this problem. Q9 is a straight forward question related to one property of CFLs. Even the students who rote learnt the property can answer the question, and we do not expect that students will use examples. Q10 checks whether students can differentiate between the problems regarding relationships between the classes of languages and relationship between the members of the languages. This question further tries to find at what abstraction level the students think.

We conducted five think aloud interviews to check motivation of students towards automata and the mental process followed while solving the problems related to automata. Table 1 provides the series of questions asked during the think aloud interviews. In addition, the students were explicitly asked why they feel it difficult to comprehend these concepts.

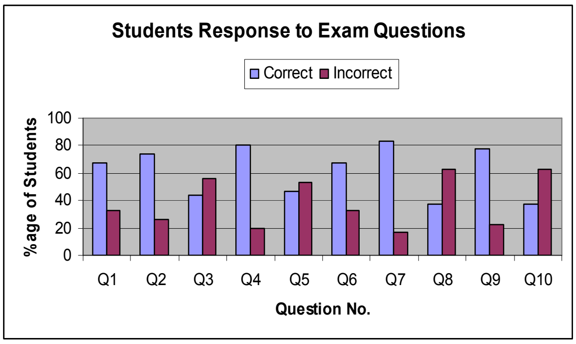

Fig. 2 Students' Response to Examination and Assignment Questions

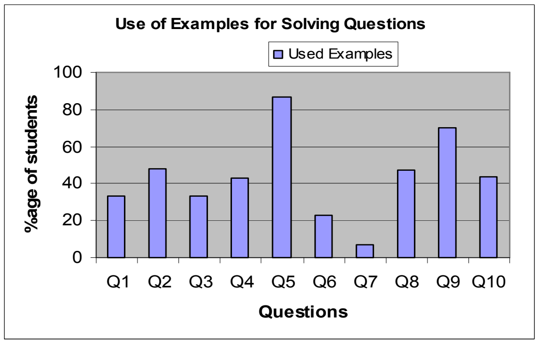

Fig. 3 Use of Examples for Solving Questions

4. Results and Discussion

The analysis of the examination results show very strong tendency to reduce the abstraction level of the problems by using examples. Some proportion of students generated concrete examples for every question. Overall questions can be divided into those for which concrete reasoning helped, those for which it hindered, and those that can be answered by rote learnt knowledge. Figure 2 shows response of students on each question. Figure 3 shows what percentage of students used examples for solving questions. Students particularly faced problems in Q3, Q5, Q8 and Q10. Approximately 61% of the wrong answers came from these four questions.

Q3, Q5, Q8 and Q10 are the questions which demand extra attention of the student and need the student to have good understanding of the underlying concept. It needs the existence and actuation of proper links in the mental concept map of the student. Bars of Q3 and Q5 in figure 2 suggest that students particularly face problems while performing concatenation. Figure 3 shows that the students who tried the Q8 and Q10 by thinking of an example, most of them reached the correct solution. For Q8 and Q10 more than 65% of the students using examples got correct answers. These are the kind of questions for which concrete reasoning helped. The students failed to provide proper reasoning when they handled them at abstract level and tried to make inference between abstract concepts instead of reducing the abstraction level. Q5 is another indication of problem to students when they do not use examples and try to make inference between abstract concepts but in this case the use of concrete reasoning hinders instead of helping the student. In such scenarios the students need to be capable of handling a certain level of abstraction. Q3 also falls in the same category Q5 does. Use of examples does not always help students reach the correct solution. It further needs proper knowledge of the underlying concept to build concrete reasoning on.

Q1, Q2, Q4, Q7 and Q9 are the questions which can be answered by rote learnt knowledge of closure properties of regular laguages. Figure 3 shows that a significant number of students tried to use examples to solve even such simpler problems. Only Q7 was answered without examples and correct solution was reached. Figure 4 shows that 100% of the students who used an example to solve Q7 got it wrong. These wrong attempts were made by students following the mental process studied by Hazzan [3]. Hazzan narrates that while handling abstract concepts, students unconciously try to reduce the abstraction level of the concept [3]. Since this was a trivial problem and a simple property of regular languages could have been used as done by most of the students. This is the kind of question for which the concrete reasoning did not help. The students who did not have good understanding of the property tried to prove their claim using an example which is a wrong technique. This brings us to another deduction which needs further investigation, that these students might even do not know that presenting an example is not the proof of the claim. So we infer that students tend to use examples without realizing that this problem can be solved without an example and the use of example hinders instead of helping the student. On basis of the data other than Q7, we further infer that students who attempt a question by first finding examples are more likely to find correct solution. Wrong attempts to Q1 were relatively higher as compared to other 4 similar questions discussed above. So we included questions related to complement of a CFL and concatenation in think aloud interviews.

We explored the mental maps of a subset of these students using think aloud protocol. The data from these flashes out the details of the kinds of reasoning we have been mentioning. We also use it to support our idea that purely abstract text book material subjectively hinders learning by lowering motivation.

Taking a detailed example first one finding of the think aloud was that whenever the students are asked to think about a CFL, they come up with the language anbn, which sometimes causes problems in solving a problem. For example while answering Q4 of table 2, four out of five interviewees thought of anbn at first. This line of reasoning was leading them to the wrong solution. But afterwards they realized that they can also pick a regular language to take union with and reached the correct solution. This result supports our hypothesis 2. The reason of thinking anbn whenever one talks about a CFL might be that the concept CFL has been closely associated with concept anbn. Another reason could be that anbn is acting as a label of concept CFL. This claim needs further research and we keep it for future work. One of the subjects involved in think aloud interviews faced very little problems in comprehending abstract concepts. This led us to a deduction that the understanding of these concepts also depends on the quality of the relationship between the object of thought and the thinking person as described by previous studies [3, 11, 12].

The think aloud interviews suggested that students fail to see any concrete application of automata concepts which is a factor hindering meaningful learning and is responsible for lack of motivation. Unless the students see something working or at least have a feeling that there exists a thing similar to what they are being taught, they can not make significant learning about the taught concepts. This result backs our hypothesis 3.

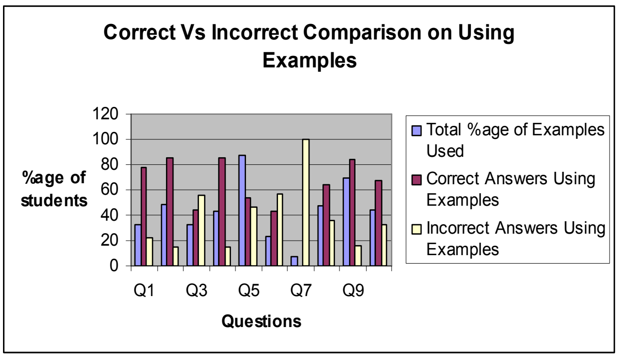

Fig 4 Use of Examples for Correct and Incorrect Answers

5. Conclusion

Learning abstract concepts is a difficult task. This study shows that students face problems while making inference between one or more abstract concepts and they find it easy to solve when they use examples to solve the questions. This paper studies the response of 30 students to exam and assignment questions. The students' response shows that whenever they try attempting the problem using an example, most of the times they end up with a correct solution. But, in order to reduce the level of abstraction of the concept, they sometimes tend to use examples where examples are not even needed. Result of Q7 is an example of such a scenario. They do so firstly, because they fail to handle the abstract concepts and secondly because they do not know when to use examples and when not. While proving something students should not be using examples, whereas if they are disproving something, they can use examples. The results of exam questions support two out of three hypotheses. Moreover, the results depict that students face particular problems handling questions related to intersection and concatenation of RLs and CFLs. This paper also summarizes the results of 5 think aloud interviews which were conducted to study the mental process, the structure of the concept map and the line of reasoning followed while attempting a problem. The results of think aloud interviews suggest that lack of motivation is a hindrance in meaningful learning of abstract concepts in automata, hence proving hypothesis 3. Briefly, students fail to make inference between two or more abstract concepts, proper links are not actuated at proper time and students feel less motivated while learning automata since they do not directly see any application of these concepts. Think aloud interviews showed that students think of anbn whenever they are told to think about a CFL. Our future direction is the study of the reason which compels them to think like that.

Acknowledgements:
We would like to thank Dr. Shahab Baqai and Malik Jahan Khan who are respectively the Instructor and TA of the course Automata offered at LUMS, to provide the data for analysis.

References:

[1] J. E. Hopcroft, R. Motwani, J. D. Ullman, *Introduction to Automata Theory, Languages and Computation, Second Edition*.

[2] C. I. Chesnevar, A. G. Manguitman, M. P. Gonzalez, M. L. Cobo, Teaching Fundamentals of Computing Theory: A Constructivist

Approach, *JCS&T,* Vol 4, No 2, August 2004, pp 91-97.

[3] O. Hazzan, Reducing Abstraction Level while Learning Computability Theory Concepts, *In Proceedings of ACM ITiCSE,Denmark*, Jun 24-26 2002, pp 156-160.

[4] M. H. Hoffmann, An Engineering Model of Learning, *In Proceedings of 35th ASEE/IEEE Frontiersin Education Conference,Indianapolis*, IN,, Oct 19-22 2005.

[5] T. J. Cooney, C. R. Hirsch, Motivation: An Essential Component of Mathematics Instruction, *Teaching and Learning Mathematics in the 1990s*, August 1990, pp 100-102.

[6] M. Ben-Ari, Constructivism in computer science education, *Journal of Computers in Mathematics & Sc. Teaching*, 20(1):45–73, 2001.

[7] R. Davis, C. Maher, and N. Noddings. (Eds.) *Constructivist views of the teaching and learning of mathematics*. Nat. Council for Teaching Mathematics, 1990.

[8] S. K. Chin, A Practical Model for Improving Student Learning of a Programming Language, *In Proceedings of 36th ASEE/IEEE Frontiersin Education Conference*,San Diego, CA , Oct 28-31 2006.

[9] URL: http://math.youngzones.org/Math_2213_webpages/vocabulary2213.html Website visited on May 8, 2007.

[10] URL: http://dorakmt.tripod.com/mtd/glosmath.html Website visited on May 8, 2007.

[11] D. C. Aharoni, S. Ergo, Cognitive Processes of Students Dealing with Data Structures, *In Proceedings SIGCSE 2000*, Austin, 2000, pp 26-30.

[12] J. Perrenet, E. Kaasenbrood, Levels of Abstraction in Students' Understanding of the Concept of Algorithm: the Qualitative Perspective, *In Proceedings of ACM ITiCSE*, Italy, Jun 26-28 2006, pp 270-274

[13] J. D. Novak, Learning, Creating and Using Knowledge, *Lawrence Erlbaum Associates Inc*. 1998.