SCENARIO BASED THINK ALOUD PROTOCOL FOR PROBING STUDENT PROBLEM SOLVING SKILLS

Benjamin Yu
British Columbia Institute of Technology
byu@bcit.ca

Wendy K. Adams
University of Northern Colorado
Wendy.Adams@colorado.edu

ABSTRACT

When interviewing students solving problems, typically researchers either follow a think-aloud protocol or ask students to describe what they are thinking as they solve problems. It is quite common for researchers to have a difficult time learning about the subject's thought processes and problem solving skills during interviews because subjects at times may not adequately verbalize their thoughts and quite often they have difficulty solving the problem – either getting stuck or going down blind alleys. In this paper, we discuss our experiences with adapting a protocol designed as part of the Colorado Assessment of Problem Solving. We used the protocol to assess Computer Science students while solving a problem in their area. This scenario based think-aloud protocol is designed to engage the subject partly as a third party observer, and partly as a problem solver herself. In the process, the subject finds herself less and less conscious of being studied and begins to immerse herself as a participant in the problem solving exercise with her own thoughts and ideas. A number of essential skills required for problem solving were identified through these scenario based think-aloud sessions so instructors may be more informed on what specific topics/skills to teach.

Keywords: *think aloud, protocol, interviews, thinking, problem solving, Physics, Computer Science*

INTRODUCTION

One of the most difficult challenges for teachers to be effective in their teaching is to be able to understand how students think and process information. Without knowing how the students relate what they have learned in the past with the way new information is processed, it is difficult to uncover how they approach a problem and solve it. In order for the teacher to steer the students in the right direction without being sidetracked by prior experiences and problem solving strategies, an effective tool to reveal how students solve problems is needed. Ericsson and Simon (1980) have shown that it is possible to study objectively the form of thinking that occurs covertly in many types of typical tasks and activities in everyday life through think aloud or talk aloud protocols (or protocol analysis). Think aloud protocol has been used in Human Computer Interface design (HCI) to understand and observe how users work with different interfaces (Stone et al., 2005) (Nielsen et al, 2002), which can be for a software application, a piece of hardware, or a task they have to perform that requires interfacing to some media. The same protocol has also been used in education research to understand student engagement in problem solving in different areas. Students are asked to verbalize their thoughts as they work through a problem.

At the University of British Columbia and University of Colorado, Boulder, we have found that while the think aloud protocol has been quite useful in some situations, it has not produced as much insight into student thinking as we had hoped at other times. In this paper, we first identity when a think aloud protocol alone fails to elicit student thought processes while solving problems. We then describe our experiences with creating a scenario around a problem for the students to solve during the think aloud session, which is similar to what has been developed as part of CAPS (Colorado Assessment of Problem

Solving) (Adams, 2007), and provide examples of how this has been used in education research in Computer Science. We will describe how this process was adopted and the results of interviewing the students.

LITERATURE REVIEW AND MOTIVATION

In Computer Science, one of the more difficult topics for students to understand is mathematical induction (Dubinsky 1986) (Polycarpou, 2006) (Segal 1998). It is a proof technique and is usually taught in first or second year. The technique is often used to prove program correctness and students learn how to apply the technique to prove that a program is computing what it claims to compute. In our research we want to learn about student thinking while solving such proof problems. Why is it so difficult for students to conduct mathematical induction proofs? How do they approach such a problem? Can they ask the right questions, plan and perform various tasks specific to this type of proof? Do they know what will be productive to try? Can they step back and evaluate their own progress?

In general, we are interested to learn about the thinking that goes on while a student is solving a new problem. Many educators use the term "problem solving" to cover a very broad selection of tasks including answering back of the chapter textbook problems. We will use the specific definition: "Problem solving is cognitive processing directed at achieving a goal when no solution method is obvious to the problem solver." (Mayer, 1992) This requires that the classification of a problem be based on the solvers response to the problem rather than the task itself. If a person is an 'expert' in their field, then it is very likely that a task that is a problem for others will only be an exercise for the expert. Therefore all of our research involves tasks that the subject is unable to quickly step through to a solution.

Think Aloud Protocol

Think aloud sessions are usually conducted in a one-on-one interview format. When studying problem solving, typically the subject will be presented with a problem to be solved and they will be asked to verbalize their thoughts as they attempt to solve it. Some guidance may be provided by the researcher at the beginning so the subject will know what is expected in the process. As an example, the researcher may illustrate the thought process of adding two three-digit numbers together by verbalizing how she would add the unit digits first, then the ten's digits next with possible carry, etc. It also helps at the beginning of the interview to make sure that the subject is comfortable with the researcher and the environment. Hence, asking the subject, if she is a student, about their background, major, and any other relationship building questions will help establish the rapport and ease the student into a mode of expressing their thoughts later on.

It is important to note that a successful think aloud session allows the interviewer to observe the subjects' *problem solving process* rather than their *opinion* or *analysis* of the solving process. The key is to have the subject verbalize their thoughts rather than changing the original intent of solving the problem to explaining *how* to solve the problem. The latter would distort how the subject actually solves the problem as they feel obligated to provide reasoning for their steps such as explaining why they choose their strategy in solving the problem. Such extra cognitive processing disturbs their normal mode of problem solving. Another problem with the 'explanation' mode is that subjects often are focused on explaining and fail to really engage with the problem. In these cases, it often appears that a subject is unable to solve a problem that they actually are capable of solving or at least making progress on.

The interviewer should allocate sufficient time and provide an environment with minimal disturbance to the subject during the think aloud session. Another important but sensitive factor is when should the interviewer speak to the subject and when should they be quietly observing the subject, especially when there is a long period of silence and inactivity. It's important to set expectations in the beginning so that the students know that they are expected to verbalize what they are doing when they can. Sometimes the subject may be deep in their thinking and be silent for a while. It is often tempting to ask the subject what is on their mind. However, even such a simple question may disrupt the subject's thought process and may not reveal what she was actually thinking at the moment. When the problem is sufficiently difficult that the subject must use all available cognition for the problem, the interviewer is very limited in what they can learn. When the subject is at a point that they can verbalize again, the subject sometimes falls into explanation mode which is typically a summary of the subject's conclusions of that thought process leaving the actual thought processes hidden from the interviewer.

Difficulties With Unscaffolded Problems In Think Aloud Protocol

While think aloud sessions can be effective in eliciting student thought process, the success rate of obtaining insightful data depends on a number of factors. The choice of a suitable problem, whether the subject engages in the problem solving process, the subject's ability to solve the problem and not get stuck, the subject's cognitive load during the solution, and the subject's willingness to speak out loud, etc.

Even with a well constructed problem, it is difficult at times to get the subject to engage in the problem solving process. Often students express that they should know how to solve the problem but have forgotten how to do it. They simply try to recall 'the solution method' that they have learned rather than approach the problem afresh and work through what they can. These attempts at recall rarely lead to a solution, and students often become frustrated and/or embarrassed that they can't do what they think they should be able to do.

At other times, subjects are either so involved with the problem, or they are not used to verbalizing their thoughts, that a number of the sessions are but silence. Even when probed to solicit some response from the subject, which is not recommended as indicated above, the subject would only be able to describe minimally what they were thinking. Still at other times, the subject may have difficulty knowing how and where to start solving the problem, and many would find it difficult to verbalize their attempts due to a number of reasons. They may feel embarrassed to articulate how they were feeling, such as incompetent or unconfident in solving the problem, not being able to recall how a similar problem has been taught in class and how to solve it, or having attempted some possible attempts but only lead to dead ends.

When students do successfully solve a problem during an interview, we still find ourselves "in the dark'. Much of student thinking is not verbalized especially when many things happen all at once. When someone is expert at a particular skill, it becomes automatic. When this happens, they often don't realize what they've done. It's similar to tying one's shoelaces. The details of the process are nearly invisible to the observer since it is often difficult to describe how to explain the process in words only.

RESEARCH DESIGN

Scenario Based Problems Used With Think Aloud Protocol

We now describe an augmented form of think aloud protocol which we have found to be quite effective in creating a comfortable environment for the subjects to solve a problem and to verbalize their thoughts. Instead of putting the subjects in a situation where they may feel that they are being observed in a study by the interviewer, the subjects are asked to observe how two other individuals would solve a problem. The setting can vary but we have found, as an example, that by putting the subject in a scenario where she is in a position to make a decision on hiring one of two interns in a fictitious company, through observation and critiquing of how the two interns work together to solve a complex problem, and how each intern approaches the problem individually to be quite effective. This relieves the pressure from the subjects having to "perform" in front of the interviewers and the focus is directed from the subjects themselves to the two interns that are solving the problem. The subjects become third party observers while they are actually being observed by the interviewer. The subjects appear to be more at ease in working through the problem, alongside the two problem solvers in the scenario, and providing commentary as they work through the solution.

To implement this think aloud session, the subjects are given a script which presents the scenario and records the conversation of the two interns who collaborated in solving the problem. Interspersed in between the dialogue of the two interns at various points are questions directed to the subjects on whether they agree on the steps taken, the arguments that the two interns have constructed / defended /

opposed as they work through the problem. From time to time, the subjects are also asked to pause from reading the script and to offer their ideas on how they would solve the problem themselves.

This form of scenario based think aloud session does confine the subjects to solving a problem in a particular route, rather than giving them the freedom to solve the problem anyway they want. Even though the subjects are asked on a number of occasions how they would solve the problem on they own, it is quite different from the traditional think aloud protocol where they are asked to solve a problem with no scaffolding or guidance. Our results in interviewing Computer Science students using this type of think aloud sessions show that the students are much more engaged than using a non-scripted problem in a think aloud session. When compared to think aloud interviews with non-scripted problems we see a lot more insights on student problem solving skills with the scripted scenario based problems.

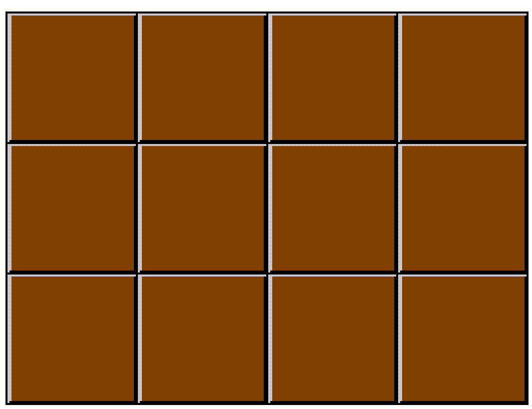
Creating a scenario is an iterative process. Once we have collected some data from the interviews based on an initial draft of the scripts, we can then modify the script to probe deeper and more specifically on what the subjects may be thinking at different points during the session.

Adoption for Computer Science

The following is the initial portion of the script we have used in our think aloud interviews for Computer Science research on how students solve mathematical induction problems. The entire script is too long to be reprinted here. The subject is asked to read the script and answer the questions (in bold) interspersed in the script. The script consists of a dialogue between the two interns, Jasmine and Sara, who are being assessed on their problem solving abilities as part of their hiring process. The subject is asked to evaluate the conversation and decide which of the interns she would hire.

The problem which the two interns are asked to solve is stated as follows:

> Assume you have a chocolate bar consisting, as usual, of a number of squares arranged in a rectangular pattern. Your task is to split the bar into small squares (always breaking along the lines between the squares) with a minimum number of breaks. How many will it take? Here is a sample of a chocolate bar to get you started:



After reading the chocolate splitting problem, Jasmine and Sara sit down and start thinking about the problem.

Jasmine thinks awhile and says, “It is difficult to think where to start. The question asks for a minimal break, so there is probably an efficient way to splitting the chocolate bar than just randomly splitting it.”

1. **Do you have any ideas about what is the fastest way to split the chocolate bar? What would you try to figure out first?** (No need to work out the whole problem right now.)

2. **How do you feel about this problem?**

    a. **Could solve it on my own right now.**
    b. **Would rather solve it in a group setting.**
    c. **Need more information before I could say.**
    d. **It's completely over my head.**

Please explain further:

I'm sure they expect a fast method, that's sort of the whole point of the problem. Don't you think? But they also ask for "how many", so they are probably expecting a formula based on the size of the original chocolate bar.

Sure. I guess you are right. How do we go from a chocolate bar to a formula then?

3. **What do you think?**

After a little discussion they agree on a plan.

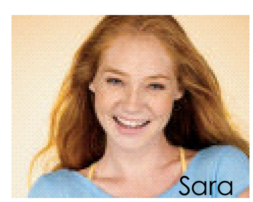
Ok, let's try to figure out how many splits we need for the sample chocolate bar presented in the problem.

4. **Is this a reasonable plan? Is there any reason that their plan may be a waste of time?**

5. **Is there something you want to do next or would you rather see what the girls are planning?**

I am losing count on how many splits I need for the 3 X 4 chocolate bar, but I think the best I can do is 11 splits.

Yeah, I have the same too. Are there faster ways to split it? Does it matter if I split a big chunk or a small chunk early?



I am not sure. Let's try some other chocolate bars of different sizes and see if there is a pattern!

6. **What do you think is the answer to the number of splits required for any m x n chocolate bars?**

7. **How confident are you about your answer?**
   a. **Positive**
   b. **Pretty sure**
   c. **Think it's close**
   d. **Not sure at all**

This is interesting. For a 2 x 1 bar, we need one split. For a 3 x 1 bar, we need two splits. For a 3 x 2 bar, we need five splits. Can we conclude that for a m x n bar, we need (m x n) – 1 splits?

Fascinating! It will be nice to do a few more samples, but the bigger the chocolate bars, the more time consuming it is to figure out how many splits. My head is starting to hurt already! Why don't we find a way to show that your theory is indeed correct without having to work through each chocolate bar size?

8. **How would you proceed from this point on? Do you think Sara's suggestion is even doable?**

etc.

Limits Of Adaption

The scenario based think aloud protocol developed as part of CAPS was initially written for studying problem solving skills in Physics students. There was a number of goals intended for the development. These include stepping the student through a likely path to a solution, eliciting feedback from the student on their problem solving process and creating the feeling that each character in the scenario is a real student. When adapting the scenario for computer science, not all the details of each of these goals was implemented since a fully developed scenario would require several iterations of tuning and refinement. However, we were very pleased to find that even the first draft of this adapted scenario was quite productive for our study of Computer Science students in their problem solving skills particularly in the area of mathematical induction.

Stepping the subject through the problem solving process to a solution is the most important aspect of this scenario and was fully implemented in the Computer Science protocol. It solves many of the typical problems identified above. The subjects who were trying to recall a solution they previously learned to solve a problem, are now immersed in a different setting, that of working with these two interns as a group to solve the problem. There's also less personal pressure of failure since much of the direction in solving the problem is provided by the interns. The size of problem solving steps chosen and the comments by the characters were designed to reduce the cognitive load for the subject. Also, what has been learned from each step is summarized by either intern before moving to the next step. We have found that this keeps the progress moving forward and the subject is able to continue verbalizing as they move through the scenario. We believe one of the main reasons for subjects to fall silent during other problem solving interviews is cognitive overload. Our experiences with this protocol support that hypothesis since all subjects have done an adequate job of verbalizing while working with the scenario based problems.

The CAPS scenario was designed to elicit a wide range of problem solving skills and processes. These skills and processes are derived from a series of interviews with a range of students. Once these skills and processes are identified, questions would be added to the scenario to encourage the students to use them. For example, to get the subjects to use the skill of estimation, they will first be asked how they would estimate the weight of an object. This will also reveal whether the subjects think of using estimation at all in their problem solving. Then the scenario will include a part where one of the characters is required to estimate the weight of the object. The subjects are then asked if this seems reasonable. With this pair of questions the interviewer will be able to determine if the subjects consider estimation useful in this case and then see if the subjects *can* estimate. The Computer Science protocol described above does not elicit nearly as many skills and process as the CAPS scenario, and the ones that it does were determined by the author's past experiences rather than a series of interviews. Nevertheless, we were encouraged to see how much more information this "time efficient" scenario brought us. It was adequate for the interviewer to find a wealth of data about mathematical induction for his purposes. It surely could provide more in depth information if it were honed based on the interview results.

A third goal of the CAPS scenario was to create believable personalities for each of the interns, Jasmine and Sarah, and to see if the subjects can pick out the difference in their problem solving skills and approaches. The first CAPS design and development interviews were not successful at having the students notice each intern as having a distinct personality. This was a problem because part of the evaluation process of the CAPS was to see how well the student can evaluate another solver's abilities. For this reason pictures were found to represent Jasmine and Sarah and extensive effort was put into creating two distinct personalities each with a consistent but different set of problem solving skills. When creating the Computer Science version of the scenario, the focus was not on creating two personalities. The Computer Science problem solving research was not focused on the student's ability to evaluate others, only on the skills of the student. There were only one or two comments by the subjects about the differences of problem solving abilities between Jasmine and Sarah. This is likely because students are not very strong at evaluating other's problem solving skills and this is in line with the results from CAPS where we find very few students are good at this particular task.

RESULTS AND DISCUSSION

Initial Results from Computer Science

Thirteen students from the January session, 2010, of CPSC 121 (Models of Computation) at the University of British Columbia participated in the study. Two faculty members were also interviewed to validate the tool. Of the thirteen students, twelve of them worked through the script. All twelve of the students also attempted the open-ended questions when they were asked how they would proceed from certain points during the script. All of them took up the full hour allocated to each interview, and only two ran out of time. The script is therefore about the right length for an one hour interview. All of the twelve students were very much engaged in the problem, either working through the problem themselves or analyzing how the two interns were solving the problem. As a way of comparison, the chocolate bar problem itself, without the rest of the script, was given to the thirteenth student to see if she could solve the problem herself in a traditional think aloud approach. This student attempted the problem with almost no verbalization of her thoughts, and gave up after 5 minutes.

Skills Used In Mathematical Induction

We now report our findings of the type of skills that students used to solve a mathematical induction problem like the one described earlier.

Problem Comprehension. While the problem as stated may seem to be clear to most students and the two faculty members, we were surprised that there were some who had different interpretations at different points of the script as intended. As an example, in the problem statement, it turned out that it was not clear to a few students whether they could stack the slices that have been broken off and split all of the pieces with one break. Such awareness of the possible interpretations of the problem and to clarify the problem before proceeding in the proof are important initial steps in deriving a correct solution to the problem.

Identification of a starting point in the problem solution. While some students may approach a problem and quickly identify how the solution can be derived or which proof method should be used, other students may not be so quick especially if the problem is not related directly to any of the courses they are taking currently. The goal of general problem solving research is to see how subjects approach a problem without having them to fall back on a solution or algorithm they have learned in class or read somewhere. In our case, the students selected for the interviews all came from a course where they learned mathematical induction. Although they were not told that the problem was a mathematical induction problem at the beginning, many quickly realized that such was the case and proceeded to apply the proof method for the problem. At other times, we have observed where students do not know where to begin, or if they do not recall seeing the problem before, they may try for a short while but give up easily. This is probably due to their lack of confidence in proceeding not knowing whether such an effort is worthwhile or not. They may attempt a solution but may also decide not to continue prematurely. We conclude that a good repertoire of identifying key features of the problem to start the problem solving process is essential. Without a possible set of starting points to tackle a problem and without a set of problem solving strategies, the subjects can easily get discouraged and give up. The students we interviewed happened to identify the problem category right away and at least they know how to proceed initially. This provides additional motivation for them to work on the problem and see how the interns would solve the problem.

This step in the problem solving is crucial because if the subjects do not feel confident in working on the problem, they will soon give up, and little can be gained from the interviews. Students would benefit by working on problems they have seen before and the use of faded examples (Gray et al, 2007) to build up this repertoire of problem solving skills before proceeding to a similar problem but cast in a totally different setting.

Attempt the solution with small sample problems. By working through the problem with some small sample sizes or parameters, the subjects may begin to see some consistencies in the solutions. The students need to choose carefully what constitutes small sample size problems or problem parameters, and be able to identify how they affect the results. This requires careful planning and observation skills. The students also need to adjust their sample sizes based on their observations so they can establish a hypothesize in solving the problem. All twelve students attempted to find the smallest number of breaks required for the sample chocolate bar given in the problem at the beginning of the session. All except two students tried other smaller sample sizes before proceeding with the script. However at a later point in the session, both of these students also tried smaller sample sizes before they attempted to come up with a general solution (at question 6 in the script).

Hypothesize from small samples to a general solution. This step requires a leap of faith from having collected some empirical data to establishing a general hypothesis. It is also a necessary step in order to proceed with the proof since once a mathematical induction proof is identified, the proof method requires a general solution in the form of a formula to show that it works for all cases. To be able to come up with such a hypothesis requires the students to be able to trust their intuition, and committed to testing out whether this hypothesis will work. In our interviews, all students came up with the general solution of requiring n-1 splits for any chocolate bar of size n near the beginning of the session. The students also need to be aware of the decision points they are making along the way, so that they may retrace their proof and try other alternatives at these decision points.

Show the base case works. Specifically for a mathematical induction problem, the students need to identify what constitutes a base case, and whether that base case works. The students should understand why a base case is needed and how it supports the development of a mathematical induction proof. The base case selected among the twelve students ranges from one to four (for the chocolate bar size). This shows that the students who did not choose chocolate bar size of one do not really understand the purpose of mathematical induction proof, since mathematical induction proofs are designed to prove a general solution for all natural numbers (i.e. 1 to infinity).

Pick a suitable meaning for "n". Mathematical induction requires that once a base case is established, and assuming that the hypothesis is applicable for a problem of size n, the next step is to establish that it applies to a problem of size n+1, and if such is the case, it will work for all n's. The identification of what n represents requires careful thought. In the problem above, the interns were distracted by the size of the chocolate bar being m x n. The use of "n" was typical in the teaching of mathematical induction proofs in class, but it is also used in the problem to describe one dimension of the chocolate bar. A number of students followed the wrong path like the interns initially and did not realize that they should have rephrased the problem in terms of a chocolate bar of size n rather than of size m x n. The ability to re-interpret the problem, or re-cast of the problem in terms of other variables could have simplified the proof tremendously.

Show that the problem works for "n+1". This is perhaps the single most challenging part of any mathematical induction proofs. Students need to be creative in coming up for a solution in this step. This often requires careful observation, ability to think "outside the box", looking at the problem from a different angle, recast the problem into another problem, etc. A repertoire of problem solving skills will help the students especially in this stage of the proof, and not only will this help them in the final derivation of the solution, but also provide them with the resource and confidence to persevere in the process.

Resolve any inconsistency in establishing result for "n+1". This step involves identifying any inconsistency and potential contradiction in establishing the result for case "n+1". The subjects have to be confident in their problem solving skills, the reasoning process they have used so far in the proof, and be able recall the possible decisions made in the process and be flexible enough to backtrack to try other alternatives. In one part of the script, the two interns seem to have done the right calculation, follow the right reasoning, but the answer they expected is different than what they got. Some of the students realized where the problem was right way by referring back to one of the previous steps. This requires careful bookkeeping of all the steps made in the process and an understanding of those steps to resolve any of the inconsistencies from the data throughout the process.

Summarize the entire process and write out the entire proof. While the students may be able to apply a mechanical approach in working through the proof, they may not be able to reflect back and write out the entire proof even after all the steps have been completed. Often times, the students may wonder at the end of the proof whether they have actually proved anything. Some of the students worked through the proofs along with the interns, but when they were asked to write out the proof, they were not able to do so. The ability to extract information from the proof activities and organize this in a concise presentation of the proof is required to demonstrate that the students have actually understood the problem fully, and articulate how the proof is conducted.

CONCLUSION

We have implemented a scripted scenario-based problem in a think aloud interview to learn more about computer science students' problem solving skills used while solving a mathematical induction problem. The scenario based think aloud protocol departs from the traditional think aloud protocol primarily by providing the subjects with guided commentary on how other problem solver(s) would have solved a problem. The subjects are asked instead to evaluate the problem solvers rather than actually solving the problem, although at different points in the interview, they are given the opportunities to solve the problem themselves. The main benefit of this approach is that the subjects are more engaged in these sessions, they are able to articulate a lot more of their thought process, and they appear to be more at ease in working with the problems and in the interviews. We have also identified some of the skills needed for students to conduct mathematical proofs in our observations. The script can now be further enhanced given the insights we have gained from these interviews so we can probe more deeply into what students are thinking in any of the particular points during the exercise.

REFERENCES

Adams, W. K. (2007). Development of a problem solving evaluation instrument: untangling of specific problem solving skills. PhD Dissertation. Department of Physics, University of Colorado. Available at: http://www.colorado.edu/physics/EducationIssues/Adams_Dissertation.htm.

Dubinsky, E. (1986). Teaching Mathematical Induction I. Journal of Mathematical Behavior. 5, pp 305 – 317.

Ericsson, K.A. and Simon, H.A. (1998). How to Study Thinking in Everyday Life: Contrasting Think-Aloud Protocols with Descriptions and Explanations of Thinking. *Mind, Culture, and Activity.* 5(3), pp 178 - 186.

Gray, S., Clair, C., James, R., Mead, J. (2007). Suggestions for Graduated Exposure to Programming Concepts Using Fading Worked Examples. *International Computing Education Research Workshop, Proceedings of the third international workshop on Computing education research.* pp 99-110.

Mayer, R. E. (1992). *Thinking, problem solving, cognition*. Second edition. New York: W. H. Freeman and Company.

Nielsen, J., Clemmensen, T., and Yssing, C. 2002. Getting access to what goes on in people's heads?: reflections on the think-aloud technique. In *Proceedings of the Second Nordic Conference on Human-Computer interaction* (Aarhus, Denmark, October 19 - 23, 2002). NordiCHI '02, vol. 31. ACM, New York, NY, 101-110. DOI= http://doi.acm.org/10.1145/572020.572033

Polycarpou, I., (2006). Computer Science Students' Difficulties with Proofs by Induction: An Exploratory Study. Proceedings of the 44th Annual Southeast Regional Confernece. pp 601 – 606.

Segal, J., (March 1998). Learners' Difficulties With Induction Proofs. International Journal of Mathematical Education in Science and Technology. 29(2), pp 159 – 177.

Stone, D., Jarrett, C., Woodroffe, M., Minocha, S. (2005) *User Interface Design and Evaluation*. Amsterdam: Morgan Kaufmann.