

Rochester Institute of Technology

RIT Digital Institutional Repository

Presentations and other scholarship

Faculty & Staff Scholarship

3-11-2005

Patterns of Conceptual Problems: Best Practices

Joseph Stanislow

Catherine Beaton

Follow this and additional works at: <https://repository.rit.edu/other>

Recommended Citation

Stanislow, Joseph and Beaton, Catherine, "Patterns of Conceptual Problems: Best Practices" (2005).

Accessed from

<https://repository.rit.edu/other/108>

This Presentation is brought to you for free and open access by the RIT Libraries. For more information, please contact repository@rit.edu.

Patterns of Conceptual Problems: Best Practices

Name: Joseph Stanislow
Job Title: NTID Support Faculty
Phone: 585-475-6861 TTY
Email: Joseph.Stanislow@rit.edu

Name: Catherine Beaton
Job Title: Assistant Professor
Phone: 585-475-4874 TTY
Email: cii@it.rit.edu

CAL-ED/IMPACT Conference
Irvine, California
March 11-13, 2005

Rochester Institute of Technology
National Technical Institute for the Deaf
Rochester, New York

Abstract

Patterns of Conceptual Problems: Best Practices

Core IT courses related to programming, database and networking are challenging for deaf (and hearing) learners. A cognitive connection between the abstract concept and the real world meaning needs to occur for learners to be able to apply what they learn. The focus of this paper is to demonstrate conceptual barriers to learning programming languages, and to suggest a 'best practices' approach to teaching and tutoring complex subjects such as Java. While the problems and practices are geared towards deaf students, traditionally, techniques that benefit deaf learners benefit any differently enabled learners.

1. Introduction

1.1. Background

Programming languages, like Java, are offered to first-year Computer Science or Information Technology students at most universities. Many students, regardless of physical ability, have difficulties understanding the concepts of programming languages because the concepts are abstract in nature. This paper offers suggestions to overcome conceptual barriers in the delivery of programming language classes. While this paper is focused on deaf students, the concepts and suggestions can be used or applied to any other differently enabled students as well as hearing students. Note: The term "deaf" will be used in this paper to refer to both deaf and hard-of-hearing students.

1.2. Perceptual Barriers

One of the challenges in educating deaf students is a misperception that they are incapable of comprehending complex issues. Due to the attitudinal and access barriers, few deaf students make it up the educational ladder to provide role models to their peers. Within the classroom, students are categorized as 'students' and 'deaf students.' One of the first challenges in educating deaf students is to have similar expectations established for ALL students. There have been numerous contributors to the Information Technology and Computer Science fields, many of whom happened to be deaf!

Hard of Hearing Father of Internet

Vinton Cerf was hard of hearing since birth. He first gained recognition for his contributions to the development of what came to be known as the internet. He is routinely referred to as "the father of the internet," because he helped to found Arpanet in the early 1970s, a forerunner of today's internet. As part of that development, he co-designed the TCP/IP protocol for internet communication. His frustration with communication with other researchers reportedly was one of the motivations for his work to develop internet communications protocols. His wife, Sigrid, was deaf.¹

¹ <http://deafness.about.com/cs/celebfeatures/a/vintoncerf.htm>

Telecommunication Modem

A deaf scientist, Robert Weitbrecht, invented TTY which is a method of coupling TeleTypewriters to PSTN in 1964. He also developed the telecommunication modem. These early TTYs are now antique, and can only be found in places such as the Smithsonian.²

Let there be light!

Thomas Edison, inventor extraordinaire, became deaf at a very early age. From telegraphs to phonographs to light bulbs, Edison had a tremendous impact on our modern world.

These pieces of information show us that deaf and hard of hearing people are capable of learning both abstract and concrete theories, and are individuals who can contribute meaningful things to society. Many faculty become frustrated teaching deaf and hard of hearing students because of the communication barriers. The method of delivering classroom information is important. Deaf and hard of hearing students do have different communication needs, but the ability of a teacher to present in multiple styles assists all students in learning and reinforcing what has been taught.

Population: Deaf and Hard of Hearing in USA

There are approximately 291,500,000*million people in America. Of this number, it is estimated that between 10% and 15% have a documentable degree of hearing loss. Historically, the figures have been closer to 10%; however, noise-induced hearing loss is becoming a serious issue for today's youth, and the number of adolescents with hearing loss related to noise pollution is increasing. Another contributing factor to the increase would be age-related hearing loss with Baby Boomers, which is not relevant to the scope of this paper.

The total population of deaf or hard of hearing people is quite high, yet is underrepresented in colleges. Traditional oral methods of delivering information put these students at a disadvantage. In Post Secondary Education, there has been sufficient research done, which shows very effective methods of delivering information for deaf and hard of hearing students.

There are approximately 70,700+ students with hearing loss aged six through twenty-one, who are served nationally under IDEA –the implementation of the Individuals with Disabilities Education Act.³

Gallaudet University and the NTID College at Rochester Institute of Technology, the two largest campuses in America which focus on serving deaf students, serve approximately 2,400 undergraduate students annually. It is further estimated that another five to eight times that number are enrolled in other two and four year post-secondary educational institutions^{4.i} Therefore, the probability is quite high that, during an average educator's teaching career, s/he will be educating a deaf student in the classroom. There are specific 'Best Practices' for improving the educational experience for deaf learners.

² <http://deafness.about.com/cs/ttysandrdds/a/ttyhistory.htm>

³ US Department of Education, 2002, p.II-20

⁴ The National Center for Education Statistics, 1999; Rodriguez, 2000

1.3. What Research Says About Deaf Learners in Postsecondary Education

A variety of sources indicate that there are common themes in teaching deaf students. Due to the aural (and sometimes oral) limitations of the deaf learner, communication presents as the predominant barrier to learning. Research indicates that deaf students have multiple visual demands; they must look at the teacher, the board, the overheads, the powerpoints, and possibly the interpreters or notetakers. Visually, the classroom experience can be exhausting. Multiple visual demands, such as when a computer screen includes simultaneous use of captions, graphics, and signing, may also impact deaf learners. A good rule of thumb for powerpoints is to use no more than three lines of text per slide.

The need for structure becomes very important for deaf learners. A good 'road map' enables deaf learners to follow the theme of classroom lectures. Students were found to be very highly 'dependent' learners, indicating that the optimal learning occurs when there is a clear structure to the material and information. Too much information delivered solely via lecture can be detrimental for deaf learners. Deaf learners have a documented need for active learning experiences; experiences that allow them to interact with materials and make the learning 'theirs.' Graphic organizers, such as KWL, concept maps, or other visual aids can support the spoken communication delivery methods.

The abovementioned techniques are also beneficial to the hearing community as well. And certainly the diverse members of the deaf group do not all share identical needs. Within specific pockets of learners, some needs are common. Students who have been deaf from birth or a very young age (pre-lingual deaf) will likely have communication barriers specific to the use of American Sign Language (ASL) or other visual languages. Frequently, within technical subjects, there are not yet standardized signs, and the interpreter and students work together to generate an acceptable sign. Many concepts in the Information Technology field are challenging to demonstrate through visual communication methods, which provide an additional barrier for deaf learners. If the interpreter does not have a clear understanding of the technology, the associated signs may convey the incorrect information. For example, in the database field, the terms 'link,' 'connectivity' and 'relationship' are frequently used. An interpreter used the ASL sign 'relationship' to convey the meaning for all three contexts. The deeper meaning behind the words was then lost to the deaf learner.

There are numerous theories for how learning happens, or what should comprise this odd notion of 'knowing.' Bloom's taxonomies demonstrate the various stages one must pass through before one obtains 'knowledge.'⁵ Learning, for any student, involves a series of stages one passes through to establish a framework: a process of building knowledge. Social Constructivism also views knowledge as a building process. It emphasizes the need for ongoing dialogue between student and faculty. In order to lay the groundwork for the foundation, one must first be aware of how much knowledge the student has on the subject, and also to check and make sure the existing knowledge is accurate. They are then guided from their current understanding to a path of greater understanding or wider understanding of the issues. The more students can share THEIR experiences, the more real the meaning becomes, and the stronger of an impact the learning will have on the students. By sharing

⁵ Fink, L. Dee "Creating Significant Learning Experiences," Wiley and sons, 2003

and dialogue, the students build a solid and strong knowledge, and can apply that knowledge to their own lives, thus becoming true problem-solvers.

Problem-solving skills are essential in the IT/CS worlds. Metacognitive skills are essential to literacy development for deaf learners. Yet it also poses a problem FOR deaf learners. Deaf students try to solve problems without developing a plan or thinking of the implications of their actions. This can certainly be said for hearing students as well, but this kind of impulsivity is consistently observed in deaf students.⁶ Mousley and Kelly define impulsivity as ‘a characteristic of cognition that results in failure to reflect on the appropriateness of correctness of an approach to a task or problem,, particularly when the task or problem is perceived to contain high ambiguity or uncertainty.”⁷

Lang, McKee and Connor have developed a chart of characteristics of effective teachers of deaf learners. After assessing enhanced student learning and achievement, the characteristics were noted in studies of perceptions of effective teaching. Effective teaching was defined as teaching at a good pace, appropriate use of visual materials, and the fostering of a friendly and caring attitude.⁸

While the chart differentiates between student and faculty perceptions, (See Appendix I) there were shared characteristics which were rated highly by both groups. They are as follows:

- *Emphasizes important information in the class*
- *Shows a caring attitude about student progress*
- *Knows the subject well*
- *Gives clear lectures*
- *Uses visual materials*
- *Uses sign language clearly*
- *Communicates expectations and assignments clearly*

While some of the characteristics (“Uses sign language clearly”) would not be of significance for non-signing faculty, the other characteristics, as shared in value by students and faculty, show that a supportive and structured classroom seems to be favored as a positive learning experience.

As previously mentioned, pre-lingual deaf have an important relationship with sign language interpreters. There are also deaf students who have learned sign language later in life and also use interpreters. The interpreter serves as a means of information for both students and for faculty. The implications for the student are that students become dependent on a third-party source of information. The potential for direct communication is

⁶ Mousley and Kelly, ‘Problem-Solving Strategies for Teaching Mathematics to Deaf Students,’ American Annals of the Deaf, Vol. 143, No. 4, 1998

⁷ Ibid.

⁸ Lang, McKee, Connor, “Characteristics of Effective Teachers,’ American Annals of the Deaf, vol. 138, No. 3

diminished. Sign language interpreters are one such means of third-party dependency. CART, C-print, tutoring and note-taking are other such examples.

It is crucial to know how to work with these intermediaries. There are specific guidelines available for working with sign language interpreters. (See Appendix) When using CART or C-Print, it is important to meet with the operators, share information and language specific terminology in advance, and remember to slow the delivery pace to accommodate for the time lag in translation of information, regardless of the chosen methodology.

For tutoring, the tutor must be kept in the knowledge loop. Tutors have the advantage of being able to meet with students individually, and can get a better sense of what the student understands to be true. Corrections can be made, and students have the opportunity to share their information about the materials, making the learning personalized, which is essential when trying to make cognitive connections in complex subject areas.

1.4. I.T. / C.S. –Specific Research

Much research and self-examination is being done within various educational institutions to determine specific criteria for outcomes in Information Technology (IT). While Software Engineering has generated a list of criteria⁹ IT and Computer Science (CS) have not yet reached that point. Many colleges and universities are involved in the IT ABET certification process, and at some point in the near future, there will be specific objectives and outcomes that are consistent within the field.

Despite the absence of formally shared outcomes, certainly an issue common to both CS and IT is the ability of the student to solve a problem. Maureen Sprankle, in her text, “Problem Solving and Programming Concepts,” states:

A knowledge of problem-solving and programming concepts is a necessity for those who work with computers, either as programmers or as users. Unfortunately, many students have greater difficulty with problem-solving than they do with the syntax of computer languages or applications. Since problem-solving concepts are similar in all languages and applications, students can learn them in one preliminary course. Then, when they move on to language and application courses, both the students and the instructor can concentrate on syntax. This arrangement not only saves time but also decreases frustration for everyone involved, and it improves the success rates of students.

⁹ www.abet.org

Clearly, problem solving is an essential component in mastering complex subjects. The challenge is then ensuring that the problem-solving examples make sense to the students. A simple change in wording or scenario makes a huge difference and cements that cognitive connection for deaf learners. Consider the Wason experiments related to task card selection. Information was presented and a problem was to be solved; in this case related to transportation methods for individuals. 25% of the subjects were able to solve the problem. When the information was restated utilizing a different scenario (but identical problem-solving logic) 75% of the subjects were able to solve the problem.¹⁰

It is this cognitive connection, this equating of problems to the individuals, the ‘making sense’ in one’s own way, that is essential for deaf (and likely all) learners. Ultimately, the one-to-one interaction happens at the tutoring level. Certainly problem-solving skills can be taught in groups or pairs, which will allow students to benefit from peer experience and alternate perspectives. Despite the evidence demonstrating the lack of effectiveness of lecturing, it is still the predominant mode of information delivery, presumably because of the ease of participation, reusability of materials, and positioning of control in the classroom, as opposed to the freer structure of discussion groups, story telling, peer programming and other modes of delivery. Lecture methods tend to be less well-received by the deaf learners, ranking slightly higher than the text as a reference, for programming languages.¹¹

1.5. Barriers to Communication

Faculty who teach deaf students, tutors who work with deaf students, and deaf students themselves have generated a number of documented barriers to communication. It can be easily inferred that the same problems exist in the Information Technology and Computer Science discipline.

- **Faculty teach a fast pace to complete all courses’ outlines in the course.** While certain objectives do have to be covered, there is a definitive time lag between the supported (C-Print, CART, interpreter) and the information for deaf students. Slowing the pace allows all students to follow and absorb the information.
- **Within IT, faculty often address concepts that are not directly or clearly related to the tangible world.** Students need to build on their own experiences, and relate the concepts to things that have meaning in their world. Use of real-world examples, through story-telling or demonstration, assists the students in grounding the information, which provides a solid platform upon which to build later concepts.
- **Faculty do not understand Deafness.** No one can ‘truly’ understand deafness unless they personally experience it. However, there are many resources which can show, through tape, reading or video, the barriers a deaf person faces. Learning about a student’s background (disability, culture, or gender perspective) can assist in

¹⁰ www.dtc.umn.edu/~odlyzko/doc/econ.psych.security.txt

¹¹ Fleury, “Programming in Java: Student-Constructed Rules”

providing a better understanding of the class, and can assist faculty in generating appropriate materials and examples.

- **Faculty do not use sign language.** Aside from RIT and Gallaudet, both of whom serve a large deaf population, most institutes are unlikely to have signing faculty. Having at least one faculty member who signs, provides a means of 'connection' between the deaf student and the department. Certainly these authors are not suggesting that every faculty member become fluent in sign language, but the authors are suggesting that cultural sensitivity be developed in dealing with students who sign.
- **First-year and second-year students demonstrate a considerable variability in problem solving that includes impulsivity, lack of transfer, breakdowns in reasoning, inability to organize and properly consider all the relevant information in the problem solution, and misunderstanding of the problem goals.** As Sprankle suggests, a class in problem-solving can go a long way in leveling the playing field and preparing all students in developing essential problem-solving skills.
- **Students are not willing to admit they do NOT understand, and when they are tested, they do poorly.** Being a 'friendly' or 'approachable' teacher can ensure that students will ask for help when they need it. While it is frustrating to see students fail, it is more frustrating to know that students are fearful of approaching a faculty member because they do not want to be labeled as 'stupid.'
- **Students often get lost on programming assignments because students get partial lectures.** Time is of the essence. Be sure that faculty provide appropriate time to deliver the entire lecture, while leaving time for Q and A. Chunking of information (meaningful blocks of information which can be built upon) is essential.
- **Students can follow the instructions of programming assignments and write the programming assignments that will work fine without understanding.** While assignments are a great way to get students to practice new skills. It is essential that they understand the skills themselves. Rote is not enough.
- **Students try to memorize the programming without understanding the concept for tests or practicum.** The use of multiple examples, different approaches and new ways to demonstrate the comprehension of the materials will help students understand the theory as opposed to the regurgitation of 'what the teacher said.'
- **Students jump into typing the code quickly without thinking first.** Whatever happened to Pseudo-code!?! If students cannot demonstrate what they expect the program to do, it will be very difficult to code! A map/plan/guide is needed. Students must be able to articulate their problems.
- **Deaf students do not receive as much information from classroom lectures as their hearing peers.** Deaf students need more time due to communication barriers. Diversity in delivery methods is a good thing to do!!!

- **Skilled interpreter vs. unskilled interpreter.** The ADA compels institutes to provide reasonable access to support services. Hiring an interpreter is not sufficient. The interpreter must have skills and competency in the subject area.
- **English is a second language for some deaf learners.** For international as well as deaf students, word choice can be confusing. English is a very hard language to master, and while proficiency is essential, there will always be misunderstandings. Be prepared to offer the information in a variety of ways to ensure the concept is understood.
- **Faculty or students may have had a previous bad experience with another faculty or student, and carry this prejudice into the new situation.** It happens to the best of us. A bad encounter taints all other experiences. Within the classroom, it is essential to be neutral and to deal with each person with an open mind. Leave your baggage at the door and tell students to do the same thing! A good learning environment is one where everyone feels free to share opinions with no fear of repercussion.

1.6. Tutoring

Tutoring provides many experiences for the students. If the tutor is deaf, s/he becomes a role model for other deaf students. In addition, a deaf tutor will likely have the ability to communicate clearly in both English and in sign language, which lowers the stress level and levels the communication barriers for students.

As mentioned earlier in the paper, certain concepts are not easily conveyed in sign language. One sign can have multiple meanings. The sign for 'link' can also be used for connection/connectivity or for relationship. Within a database class, these three terms; link, connection/connectivity and relationship, have very different meanings. A tutor can assist in making the communication flow better between the concepts taught by the faculty and the reception of the concepts (and clarification of said concepts), for the students.

Tutors can use multiple examples to convey the concepts. From the list in section 1.5, it is clear that real-world examples are essential in forming a good understanding of concepts. Tutors may be able to reach students on a more friendly level, and appear more approachable than the faculty member. If a concept is very abstract, a deaf tutor will be able to paraphrase and make the abstract concept have meaning for the deaf students.

Tutoring provides one-to-one, non-threatening communication, and gives interactions that are not always possible within a classroom. A variety of real-life examples or analogies can be given (such as the concept of 'arrays' and the example of a dry cleaner storage rack to demonstrate arrays) to help make that mental connection for the student. Time is not as critical in a tutoring situation. While students should be encouraged to ask questions within the classroom, and to contact the faculty member for assistance, tutoring provides a means of obtaining a different perspective for a problem.

1.7. Interpreter

For institutes who use interpreters, this section addresses best practices for supporting deaf students in CS/IT.

To improve the communication process in a classroom setting, interpreters have specific needs. There are language differences, both between sign language and English, but also concept-specific phrases which provide language differences for the interpreter.

Interpreters are bound by a Code of Ethics. It is a good idea to read and understand what they can and cannot do in a classroom setting. Each situation means a different set up, expectations and responsibilities for an interpreter. Meet with the interpreter before class begins to clarify what you want to happen in your class. The interpreter will also negotiate needs with the students.

Provide as much information as you can to the interpreter in advance. Hand-outs, power points, copies, readings, texts, videos, code snippets, definitions of acronyms, ALL information that you can give, you should give!!! If you use an email system, be sure to copy the interpreter when class is moved or cancelled. If you post materials online, allow the interpreter access. If students post materials, such as deliverables that they will be presenting to the class, to a drop box, grant access to the interpreter so that s/he can preview the presentations in advance.

In the classroom, be cautious about the use of pronouns. Remember that the deaf students are looking at the interpreter, and when you say "These are different from those" the deaf student sees the signs for 'these' and 'those' but has no idea what you were pointing to. If you talk to the interpreter in advance, slow your pace, provide appropriate advance materials, many of the communication barriers in the classroom can be reduced.

1.8 Cognitive Connections for Information Technology and Computer Science

Within the IT and CS worlds, language and terminology changes rapidly. Making the connection from a virtual world to a tangible world for deaf students can sometimes be challenging. Within the classroom, teachers can modify their teaching approach, as well as coordinate with support services, as previously discussed in this paper. However, there are specific approaches that lend to ensuring the deaf students make that mental shift from global meaning to personal meaning.

Classroom Discussion: Classroom discussions can allow deaf students to interact with hearing students and obtain a peer perspective.

Tutoring and One to One Discussion: Since deaf students do not receive as much information from classroom lectures as their hearing counterparts, tutoring is one support service offered to accommodate this problem. The use of deaf tutors to support deaf learners is beneficial in a multitude of ways: the deaf tutor serves as a role model, as well as serving as an information source. No interpreter is required, which means there

is no third-party potential for misunderstanding. Tutors can be experts or enablers; research shows that tutors who provide scaffolding environments are most beneficial.¹²

Class Environment: A studio model classroom (half lecture, half lab) supports the notion of apprenticeship. Faculty can model the appropriate process, and students can follow visually. After the demonstration is completed, students can try the exercise themselves. This model supports the concept of scaffolding as being the most important component of learning.

The I.T. department at RIT has experimented with various models. For the purpose of teaching programming concepts, students have been paired, students have been expected to learn independently, they have experienced lectures and a combination of lectures and demonstrations have been utilized. The studio model seems to provide the most flexibility to accommodate all learning styles.

1.9 Solutions

There are numerous models for learning programming languages, and there are several common themes within each of these models. The authors of this paper recognize the importance of problem-solving, and suggest the following methodology to ensure that you have success with deaf learners. Details and examples are provided in various appendices, as they relate to problem-solving approaches and models specific to programming concepts.

- Step One: **Know thy user!** Know who your students are. Do a pre-test to get an understanding of what your deaf students understand about problem-solving. Ensure that their current understanding is accurate.
- Step Two: **Be a learner!** Be receptive to student comments. Encourage peer discussions and analyses of problems; frequently the diverse backgrounds of your students may lend to new approaches for comprehending complex subject material.
- Step Three: **Be diverse yourself!** Present various examples in a variety of ways. Repetition is GOOD!
- Step Four: **Perform!** Walk through examples step by step, slowly. Allow for translation time if you are using a form of interpretation.
- Step Five: **Determine Success!** Don't assume they have learned. Test it. Verify it; either verbally by asking questions or by a class assignment or in-class quiz.

We can extrapolate the theme of ensuring success for deaf learners and apply it to the bigger picture of deaf learners in a complex content environment. Therefore,

¹² De Grave WS, Dolmans DH, van der Vleuten CP, "Profiles of effective tutors in problem-based learning: scaffolding student learning." Med Educ. 1999 Dec;33(12):901-6.

- Step One: **Identify the problem.** The deaf student should be able to verbally document the problem, in precise detail. Pseudo-code generation is a great idea. Pseudo-code will show understanding of how to solve the problem.
- Step Two: **Suggest various solutions:** The deaf learner should be able to recognize more than one way to solve the problem.
- Step Three: **Pick the 'best' way:** The deaf learner should be able to look forward and project what s/he thinks is the BEST solution for the problem.
- Step Four: **List the details of the best solution:** Mentally walking through the problem, the deaf learner should be able to predict the outcome. Doing actual code-development and examining the results is also a possibility at this step.
- Step Five: **Determine success:** The deaf learner should now evaluate the solution. Did it work? Why/not? Could something have worked better? Why/not?

In both cases; the guidelines for success in the classroom, and the means of solving problems for deaf learners, the themes are consistent: know where you are starting, know what you are doing, be willing to be flexible, and assess your progress. These suggestions sound self-explanatory, but in the classroom environment, these steps are frequently overlooked due to time constraints or assumptions. Each one is crucial for the deaf learner to comprehend, make the concepts meaningful to the learner, and to be able to demonstrate the knowledge in a different way as proof of learning.

Summary

Foundations and the provision of solid foundations are essential. Delivery style is important for laying a strong foundation, as is access to communication for people with disabilities. There are existing models for the establishment of a strong foundation within Information Sciences for abstract concepts. Problem Solving with Six Steps is a good methodology for developing the concept and algorithm structure. This could be followed with Logic Problem with Five Steps for developing the logic structure. By the time students develop the knowledge of the concept, algorithm, and logic structures they will have a better understanding of programming concepts.

Most or all modifications that faculty undergo will ALSO benefit hearing students. Speaking clearly and slowly, being approachable, empowering the students and making the programming examples real to the students are all things that benefit everyone. Finding a way of relating the IT world to the student experience allows the students to process the information in a way that is meaningful and useful.

To provide variety in problem-solving activities, related to improving problem-solving skills:

- Teachers (or knowledgeable, "advanced" students) can provide a guided problem-solving experience for individuals or small groups of students by supplying cues, suggested steps, and questions at appropriate times in the analytic process.
- Either prior to or as part of solving a problem, the teacher should involve students in an interactive/feedback situation (i.e., pairing with other students, aides, or teachers) in which they are required to explain sample problems and answer questions.
- The teacher should have the students think about, explain, analyze, and summarize problem-solving tasks with sign language, verbalization, written text, and role-playing.
- The teacher should have students "create" or construct different problem situations similar to the problem task at hand to demonstrate their understanding of that type.
- The teacher should have students conceptualize (or visualize) several different approaches to sample problem-solving tasks prior to solving an actual problem.

These suggestions are intended to help students develop greater flexibility through varied practice with problem solving. The internalization and application of new knowledge and skills is enhanced by repetitive practice, active participation, interactive discussion, and evaluative feedback.

To provide a dynamic teaching style,

0. Discuss WITH your deaf students how you can modify strategies to meet their needs
1. Discuss these perceptions with students and research particular characteristics of effective teaching
2. Be willing to modify your teaching style for deaf students (specifically slowing the pace)
3. Write the objectives on the board before the class begins
4. Be willing to experiment with what works best within the realm of each class. Every learner is different.
5. Ask questions of the students to assess their understanding. Answer questions to ensure they have the correct concept
6. Take the time to ensure they do understand, and then quiz them informally by saying "Tell me how this works" while the student is being assisted
7. Connect programming to the real world and programming
8. "Walk with students" on some examples to get their perspective
9. Take the time to ensure students are understanding
10. Use the white board/ chalk board as a means of slowing the classroom pace
11. Use visual cues
12. Demonstrate programming (make it run) before lecturing on the theory
13. Encourage "thinking before doing" to students (pseudo-code)
14. Uses a variety of strategies to reach the diversity of the student body
15. Empower the students to 'do!'

16. Observe tutoring sessions/ open tutoring hours to encourage and mentor students
17. Be flexible, willing to help, warm, and friendly!
18. Learn a few basic signs for communicating with deaf students!

There is no 'one' way to teach. The key is to be flexible and acknowledge that some methodologies do not benefit deaf learners. They also probably don't benefit hearing learners! Sit in on a two hour lecture taught by someone else, and try to summarize the experience afterwards. Students face lectures every day for the duration of their stay at your college. The more we improve communication, the easier the materials become for all students. While initial instinct will be that the 'brighter' students become bored when the pace slows, the reality is that we learn by doing, by repeating and by participating. By enabling the deaf students, we are providing a good educational experience for ALL students.

These suggestions are intended to help students develop greater flexibility through varied practice with problem solving. The internalization and application of new knowledge and skills is enhanced by repetitive practice, active participation, interactive discussion, and evaluative feedback. Deaf learners are equally as able as hearing students to master complex concepts. The communication barriers add to the challenge of learning. Ensuring that you know about your student, and his or her ability, and then building on that knowledge in a fashion that makes sense for the deaf student, is essential in scaffolding and building a solid problem-solving, logical approach to complex subjects such as programming languages!

Appendix 1

Ranking the Characteristics

The differences in perceptions found between faculty and deaf and hearing⁽¹⁾ college students are in their views of the importance of certain characteristics of effective teaching indicate a need for ongoing dialogue.

Perceptions of Ranking the 10 Characteristics:		
Rank:	What Faculty thinks are important:	What Deaf and Hearing⁽¹⁾ College student thinks are important:
1.	<i>Knows the subject well</i>	<i>Knows the subject well</i>
2.	Enjoys teaching	Uses visual Materials ⁽²⁾
3.	Uses sign language clearly	<i>Understands deafness, deaf people, and deaf culture</i>
4.	Encourages students to learn independently	<i>Communicates expectations and assignments clearly</i>
5.	Involves students in learning activities	Uses sign language clearly
6.	<i>Challenging Students' thinking</i>	Lectures at a good pace and makes sure students understand ⁽²⁾
7.	<i>Understands deafness, deaf people, and deaf culture</i>	<i>Challenging Students' thinking</i>
8.	Shows a caring attitude about student progress	Emphasizes important information in the class
9.	Is well organized	Is friendly and easy to talk to
10.	<i>Communicates expectations and assignments clearly</i>	Has a good sense of humor

Bold and Italic words mean that the faculty and students agreed the four characteristics considered most important out of the 10.

⁽¹⁾ In reviewing the results of Feldman's synthesis, both hearing and deaf college students appear to agree about the importance of most of the characteristics ranked highly by the deaf students in our study. Lecturing at a good pace may be valued more highly by deaf students than by hearing students, however, and the enjoyment of teaching by faculty appears to be viewed as less important to deaf college students. Two characteristics, *understands deafness, deaf people, and deaf culture*, and *uses sign language clearly* are unique to the population of college students on which our study focused.

⁽²⁾ In contrast, the students ranked *uses visual materials* 2nd, but that characteristic did not appear in the top 10 for the faculty group. Among students, *lectures at a good pace and makes sure students understand* ranked 6th and *emphasizes important information in the class* was 8th, but neither of these characteristics appeared in the top 10 for the faculty group.

Appendix 2

Problem Solving with Six Steps

General Problem-Solving Concepts (*Algorithm and Flowchart Structures*)

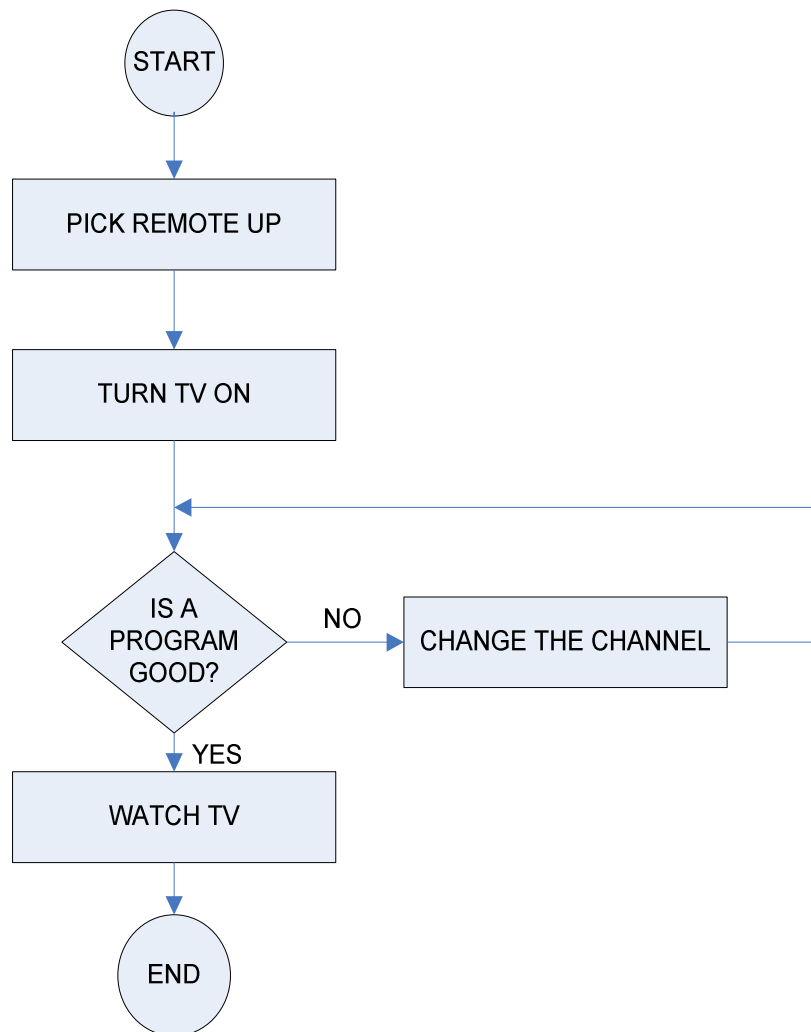
1. *Identify the problem.*
 2. *Understand the problem.*
 3. *Identify alternative ways to solve the problem.*
 4. *Select the best way to solve the problem from the list of alternative solutions.*
 5. *List instructions that enable you to solve the problem using the selected solution.*
 6. *Evaluate the solution.*
- Also, do the flowchart from Step 5.*

Here is the example:

You want to find a good program on TV.

1. *Identify the problem.*
Finding a good program on TV.
2. *Understand the problem.*
Watch a good program on TV.
3. *Identify alternative ways to solve the problem.*
 - a) Check TV guides to find a good program.
 - b) Change channels until a program is good.
4. *Select the best way to solve the problem from the list of alternative solutions.*
 - b) Change channels until a program is good.
5. *List instructions that enable you to solve the problem using the selected solution.*
(Prepare the steps.)
 - a) Pick remote up
 - b) Turn TV on
 - c) Change channels till a program is good
 - d) Watch TV
6. *Evaluate the solution.*
If TV is not working, it won't work
Can't find remote

FLOWCHART from Step 5:



Appendix 3

Logic Solutions with Five Steps

Logical Problem (*Logical Structure with basic OR, AND, and NOT logic only*).

1. *Goal: Figure out what you want.*
2. *Requirements: What's required to get it? What has to be TRUE?*
3. *Assign variable letters or names: Assign a letter to each condition, and one to what you want.*
4. *Get the equation: Set up an equation with what you want on the left, conditions on the right.*
5. *Make a truth table: That represents the equation. (The variables for conditions make the 1st columns, what you want is the last column to the right.)*

Here is the example:

You are going to the amusement park and want to have a good time. There are 3 rides there; the roller coaster, the ferris wheel, and the bumper cars. Your friend, Louise, wants to go on everything. You hate the roller coaster. In what situations would you have a good time?

1. *Goal:*

I want to have a good time with Louise.

2. *Requirements:*

Louise and I go together on *all* rides except the roller coaster.

3. *Assign variable letters or names:*

x = Good Time

r = Roller Coaster

f = Ferris Wheel

b = Bumper Cars

4. *Get the equation:*

$x = f \text{ AND } b \text{ AND NOT } r$ (Tested?)

5. *Make a truth table:*

r	f	b	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

(Row is AND and Right Column is OR)

Equation from the truth table: $X = \text{NOT } r \text{ AND } f \text{ AND } b$

Appendix 4

Example: The Lesson

Write the objectives on the board before the class begins:

Loops	How to write the loop
	How the loop works
	Real-world application of a loop

Example: The *for* Loop

Walk the students through the concept of a loop. Explain the loop with three conditions by writing the white board with step-by-step method. Show the *for* loop flow chart. Once students understand the loop and flowchart then students will be able to understand the loop statements which are the *while* loop and the *do-while* loop the rest.

First, demonstrate to run the *for* loop programming to see it as the concept in action, then the following shows how it works with the loop structure.

```
for(int i = 0; i < 5; i++)
{
    System.out.println("I will not throw paper airplanes in class");
}
```

The *for* loop has three conditions:

1. "int i = 0;" is where the value of "i" initializes at 0. Once it initializes then ignores this condition afterward. (the value can initialize at 2 instead of 0.)
2. "i < 5" is a test condition if "i" is less than (<) than 5 then it becomes true otherwise it becomes false. If it is true then it proceeds to print "I will not throw paper airplanes in class" and continue the loop till the loop stops printing when it is false.
3. "i++" is updated. It is incremented "i" by 1 each time through the loop

Example: The Array

Most students have trouble understanding how the array works and faculty need to take some time for demonstrating how the array works.

A real-world example for an array could be the dry cleaner experience. When you go to a dry cleaner, your clothes are stored on a rack that revolves, with specific placeholders. The conveyor is like the loop. The ticket number is an index. When the clothes get dropped off, the customer receives a number. After the clothes are clean (action performed) the cleaner pushes a button to place the clothes back into the same index number (number slot) that the customer has been given.

```
Example: //initializes all 0 values in the array
for (int i = 0; i < 200; i++)
    array[i] = 0;
```
