

The *UNLOCK* System: Enhancing Problem Solving Skills in CS-1 Students

T. Beaubouef

R. Lucas

J. Howatt

Department of Computer Science
Southeastern Louisiana University
Hammond, LA 70402 USA
{tbeaubouef,rlucas,jhowatt}@selu.edu

Abstract

We typically lose 50% of our CS-1 students because they possess no fundamental problem solving skills. We are developing an interactive learning tool, *UNLOCK*, to teach the students those fundamental skills. This article describes *UNLOCK*. Our goal is to increase CS-1 retention, thereby increasing our numbers of CS graduates.

Introduction

For the past several years we have been struggling to reduce the 50% attrition rate in our CS-1 and CS-2 classes. We tried changing our teaching approach, programming languages, textbooks, and assignments, to no avail. Then we realized the obvious—the majority of our students were lacking fundamental problem solving skills. It wasn't that they could not solve computing problems; it was that they could not solve problems in general!

The ability to solve problems is at the heart of computer science. Some people have a talent for it; their problem solving ability is innate. Others, including most of our entry-level majors, must develop these skills. A review of the literature reveals many approaches to learning problem solving. Some view problem solving as an art [1, 3, 5, 6,], where creativity is one of the most important traits. Others argue that certain logic and thinking skills must be learned and applied to problem solving [7]. Still others focus on developing cognitive skills, self-awareness, and motivation [4]. We hope to combine all of these ideas in teaching our students.

We believe that most students can succeed in CS-1/CS-2, after being taught the fundamental problem solving skills that they lack. Thus, we are developing our *UNLOCK* System to teach those skills. *UNLOCK* is a collection of game-like lessons, tied together with the idea of obtaining “keys” to knowledge. The goal is to master all of the problem-solving concepts, thereby collecting all the necessary keys and “winning” the game. We believe that use of this system will result in students who are better prepared to think critically, solve problems, and write programs.

In the following paragraphs, we describe our approach to designing *UNLOCK*, give examples of lessons that present different problem-solving approaches, discuss issues related to the actual presentation of the lessons to the students, and conclude by discussing the logistics of developing and implementing the system.

Preliminary Design

UNLOCK's various learning modules are grouped by categories into units. Each unit is related to mastering a particular skill or concept. When the student has mastered the unit, he [please read “he” to mean “he or she”] will be given a “key” for that unit. The student then “has the key” to solving problems where that skill is required. This key can be used in later modules to obtain hints in solving other types of problems, if that skill is required for the particular problem. When the student has earned all of the different types of keys, he will be exposed to problems that are more complex. These problems will involve a combination of several of the skills learned from the individual modules. Upon successful completion of these problems and a “final exam” covering concepts learned, the individual will earn a “master key”. He is now properly equipped to unlock the secrets of real world problems.

We initially refrain from presenting mathematics and computing problems, so as to not scare away the math-phobics. Later, we ease in math concepts so the students will begin to “do math” without realizing it. Then, we segue into computing problems and begin teaching elementary analysis and design.

Because *UNLOCK* is to serve as user-friendly educational software, it is being designed with students in mind. They should be able to learn the system quickly, and enjoy it as much as possible. The graphical user interface will be simple and the progression through the modules logical.

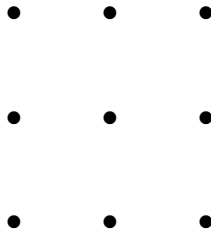
To Appear in SIGCSE Bulletin--Inroads

In the next section we present four sample lessons that can be found in *UNLOCK*. We discuss the key concept for each and explain how the problem solution relates to the concept.

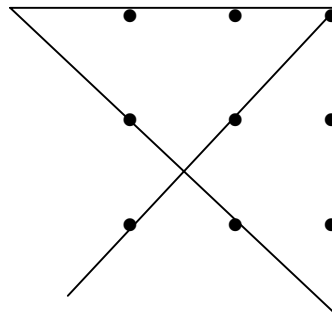
Sample Lessons

Key 010: Do not add additional constraints to the problem.

PROBLEM: *Connect the dots below without lifting your pen from the paper. Use only 4 straight lines.*[1]



Many of us have seen this problem in puzzle books. People who fail to solve this problem often make the false assumption that the lines must each begin and end on a point. This restriction was not stated in the problem. It simply formed in the reader's brain, probably based on their experience with other "dots and line segments" they have encountered. This is one solution to the problem:



If a student fails to solve the problem, he can request a hint. In this example, the first hint might be "Lines are not required to start and end on points." If the user still requires help, he can request further hints, one at a time, until he either solves the problem or depletes the hints and requests the solution.

When the student solves the problem, *UNLOCK* will present additional information related to the concept being learned. The student must carefully read the information; he will be tested on it at the end of the module.

After the student solves the problem and studies reinforcing material about the solution, he will be given other problems in this category. After correctly solving all the problems in the module and correctly answering the "test what you have learned" type questions, the student will be given a "key" to add to his key chain. This key will enable the student to gain hints in later problems related to the skills learned for this key.

Another typical stumbling block is attempting to use facts that look important, but really are not. Many students seem to have learned that all the numbers given in a problem are used somewhere to get a solution. However, the students may not have actually understood such problems, but instead memorized formulas and simply determined which value to plug in for each variable.

Key 047: Identify those facts that are useful and disregard the rest.

PROBLEM: *Little Red Riding Hood wants to take some flowers to her grandmother who is ill. It is now 3:00 p.m., but it does not get dark until 7:00 p.m. She must make it to her grandmother's house and back before dark. So Red*

To Appear in SIGCSE Bulletin--Inroads

quickly grabs her red cloak and heads outside to pick some pink monarda, her grandmother's favorite flower, which is also a favorite flower for bees. She then begins her trek to Grandmother's.

As Red begins walking, she notices a bee on her bouquet. It flies back to her home garden, and then flies back to Red. It continues to fly back and forth from Red to her home for the entire journey. This bee is very fast—it flies at a constant speed of 30 mph. Because the walk is boring, Red decides to count the number of paces it takes to get to grandmothers, and to use this to figure out the distance to Grandmother's. She knows that she is exactly 5'4" and that her step is 2 ½ feet long. This makes every two steps, or one pace, equal to 5 feet. Soon Red gets to a steep hill about halfway to her Grandmother's. Red struggles up the hill, noting that her steps are probably not 2½ feet long anymore, but shorter. When she gets about 60% up the hill, she slips, and tumbles to the bottom. With determination and renewed energy she again tackles the tall hill, and soon finds herself descending the other side. In all the confusion, she forgot her pace count, but estimated that it must be about 3000. She arrived at her grandmother's house at 4:30 p.m., just as the little bee landed on her bouquet for the last time, exhausted. How many miles did the bee travel since Red left her house?

Obviously, facts such as the grandmother's favorite flower are of no consequence in this problem. However, neither are the facts related to Red's paces, the terrain, nor the time it gets dark. To answer the question, we need only to know the speed of the bee (30 mph) and the time that elapses for Red's trip (1½ hours). From these we can calculate that the bee traveled 45 miles.

Misdirection can occur when a reader makes false assumptions while reading the problem. In this example, a reader may incorrectly assume that the question will relate to computing the distance between Red's and her grandmother's houses, which will lead the reader to begin thinking in terms of paces and distance before actually reading the question. The following example illustrates this concept further.

Key 029: Don't limit your thinking to what is typical. Consider all possibilities.

PROBLEM: *An alien spaceship hovered over a fairly flat area on the earth. Alien1 spoke to Alien2 in their alien language about exploring this strange planet. They decided to land the ship and go for a little excursion, since no Earthlings could be seen. Since they were facing south, they decided to head first in that direction. They had walked for two kilometers, when Alien2 had had enough of Alien1's ceaseless chatter. He suggested that they might cover more terrain if they split up. Alien1 would go due east for two kilometers, then due north for two kilometers, then due west for two kilometers, ending up back at the ship. Alien2 would go due west by the same amount, then head due north for two kilometers, then due east until he arrived at the ship. Since it had only taken about 15 earth minutes to get this far, Alien2 figured that he would have about 45 minutes of peace and quiet away from Alien1. He was not so fortunate, however. In thirty minutes the aliens were both back at the ship. How did this happen?*

When most of us think about the cardinal directions, we think of the compass rose. If we travel the same distances south, then east, then north, then west, we will traverse a square. However, the curvature of the earth can render this reasoning invalid, especially as the "square" gets close to the poles.

Again, if the student cannot solve the alien problem, *UNLOCK* will present hints derived from the discussion above (hints about Santa or polar bears would be used only when the student has exhausted all other hints).

The next example [2] is one of the many types of logic problems that can be found in puzzle books and on standardized tests. Albert Einstein, the author, believed that less than 2% of the people in the world could solve it. Many students do not know how to begin, but after being instructed in keeping track of 'knowns' and 'unknowns' with systematic record keeping, they can learn to solve these types of problems.

Key 031: Use known facts and logic in a systematic fashion to derive additional facts.

PROBLEM: *Use the FACTS and HINTS below to determine which of the five men has the fish as his pet.*

FACTS:

1. There are 5 houses (along the street) in 5 different colors: blue, green, red, white and yellow.
2. In each house lives a person of a different nationality: Brit, Dane, German, Norwegian and Swede.
3. These 5 owners
drink a certain beverage: beer, coffee, milk, tea, and water,

To Appear in SIGCSE Bulletin--Inroads

*smoke a certain brand of cigar: Blue Master, Dunhill, Pall Mall, Prince, and blend,
and keep a certain pet: cat, bird, dog, fish, and horse.*

4. *No owners have the same pet, smoke the same brand of cigar, or drink the same beverage.*

HINTS:

1. *The Brit lives in a red house.*
2. *The Swede keeps dogs as pets.*
3. *The Dane drinks tea.*
4. *The green house is on the left of the white house (next to it).*
5. *The green house owner drinks coffee.*
6. *The person who smokes Pall Mall rears birds.*

More hints and the solution can be found in [2].

Development Tools

We are developing lesson modules using Toolbook, Photoshop, ImageReady, and Flash (disclaimer: we have no financial or other similar interest in any of these products). For example, in creating the presentation for the nine-dot problem, we used Toolbook to provide a multiple-page, interactive lesson to present the problem and prompt for a solution. The first page displays a menu to provide navigation to any other page within the module. Page two states the problem and displays the nine dots. Page three, the most complex (see the next paragraph), not only displays the nine dots, but also allows the problem solver to draw lines to connect them. In this page, when the student draws a line, he will not be able to erase it. This forces a decision about a possible solution before starting. Entering a different solution requires starting anew (going back to the menu to return to page 3). If the student fails to find a solution, he may advance a page to receive a hint and then return to the problem. Additional hints are provided on succeeding pages. The final page of the module provides one or more solutions. A summary page describes the appropriate problem-solving approach and any “rules” or heuristics that the student should remember.

Page 3, which contained the interactivity, was created using Photoshop's graphic capabilities. It was then imported into Flash to provide the interactivity for line drawing. In the final page, Photoshop and ImageReady were used to animate the solution. Thus, by combinint tools that allow easy integration of components, we were able to quickly implement what we hope will be effective lessons.

For future modules, we plan to use Authorware, Director and other support software to create more exciting and entertaining modules by including additional graphics, video, and audio. This will make the modules more “game like” and should create more interest in pursuing them.

Conclusion

Many students enter college these days lacking fundamental problem solving skills. We are designing *UNLOCK* as an interactive, educational tool to instill and enhance problem solving skills in those students. In this paper we have discussed the motivation behind and basic design of the system, and presented a small sampling of the various problem-solving lessons. The modules of *UNLOCK* present a suite of problem solving techniques that can be used as a basis for learning software analysis and design.

In the next academic year, we will test this system on introductory computer science students. Our immediate goal is to use *UNLOCK* to provide students with needed, fundamental problem-solving skills. By satisfying this goal, we hope also to achieve our larger goals of retaining more computer science majors in CS-1 and CS-2, and of eventually producing more computer science graduates.

Acknowledgement

This research was supported in part by Louisiana Board of Regents grant LEQSF(2000-02)-ENH-UG-49.

References

- [1] Ackoff, Russell, *The Art of Problem Solving*, John Wiley & Sons, New York, 1978.
- [2] “Einstein’s Problem,” from <http://www.davar.net/math/problems/einstein.htm>, December, 2000.
- [3] Fobes, R., *The Creative Problem Solver’s Toolbox*, Solutions Through Innovation, Portland, 1993.
- [4] Levine, M., *Effective Problem Solving*, 2nd ed., Prentiss Hall, New Jersey, 1994.
- [5] Nadler, G., S. Hibino with J. Farrell, *Creative Solution Finding*, Prima Publishing, Rocklin, California, 1999.

To Appear in SIGCSE Bulletin--Inroads

- [6] Nadler, G. and S. Hibino, *Breakthrough Thinking*, Prima Publishing, Roseville, California, 1998.
- [7] Whimbey, A. and J. Lochhead, *Problem Solving & Comprehension*, 6th ed., Lawrence Erlbaum Associates, Publishers, New Jersey, 1999.