A National INCLUDES Alliance Effort to Integrate Problem-Solving Skills into Computer Science Curriculum

Ann Q. Gates
Department of Computer Science
University of Texas at El Paso
El Paso, TX USA
agates@utep.edu

Elsa Q. Villa
Center for Education Research and Policy
University of Texas at El Paso
El Paso, TX USA
evilla@utep.edu

Sarah Hug
Alliance for Technology, Learning and Society Institute
University of Colorado Boulder
Boulder, CO USA
sarah.hug@colorado.edu

Christina Convertino
Department of Teacher Education
University of Texas at El Paso
El Paso, TX USA
cconvertino@utep.edu

Johannes Strobel
School of Information Science & Learning Technologies
University of Missouri
Columbia, Missouri USA
strobelj@missouri.edu

***Abstract*— This Innovative Practice Work-In-Progress paper elucidates the approach of the NSF-funded CAHSI INCLUDES Alliance for creating change in students' competencies by an effort across eight institutions to support the delivery of one- and two-credit hour courses for three levels of problem solving in Computer Science: general problem solving, computational thinking in problem solving, and algorithmic thinking in problem solving. The courses were developed to address industry's need for improved problem-solving skills, incorporating consistent, deep collaboration with Google technical staff. The first of its kind for CAHSI, the problem-solving courses are fewer credit hours than typical courses in order to fit within a traditional curriculum. The intent is to instill complementary problem-solving, computational thinking skills, and logical reasoning needed to succeed in computer science, and make this content available across different student populations at various stages in their academic pathways. Advanced problem solving prepares students for competitive interviews. The courses create opportunities to learn across academic levels, and create new student communities, mentorship opportunities, and social connections to support retention. The paper reports on the course design, student reflection, assessment and evaluation, and an ethnographic study of the courses.**

Keywords—problem solving, computational thinking, computer science education, reflection

I. Introduction

Professors from the CAHSI (Computing Alliance of Hispanic Serving Institutions) INCLUDES Alliance, a National Science Foundation (NSF) INCLUDES alliance, created three problem-solving (PS) courses. The initial motivation for creating the courses came from a faculty recommendation at a University of Texas at El Paso (UTEP) computer science faculty retreat. The courses were developed with CAHSI faculty involvement and consistent, deep collaboration with Google technical staff and software engineers, who contributed problems and challenges combined with remote coaching of faculty and feedback to students.

The courses include PS1: General Problem Solving (1-credit hour); PS2: Computational Problem Solving (1-credit hour); and PS3: Algorithmic Problem Solving (2-credit hours). These were designed to allow more flexibility in degree plans and build students' knowledge in problem solving. The PS1 and PS2 courses were created to connect first- and second-year students, in particular females, with the major and to provide strategies for success in computing courses. PS3 focuses on skills development needed, in part, for successful interviews with competitive companies that require solving problems in real time.

II. Essential Elements of the Courses

The course design includes three essential elements: Problem-Based Learning (PBL), the IDEAL framework, and teaching methods using a cognitive apprenticeship framework. PBL, a proven instructional model [1], incorporates real-life situations and active learning strategies mirroring problem solving of professionals in the field [2]. The IDEAL framework, a structured process to problem solving, has five distinct process steps that are iteratively, not linearly, used: 1) **I**dentify the problem; 2) **D**efine the problem by thinking about it and sorting relevant information; 3) **E**xamine the options and potential solutions; 4) **A**ct on a plan and strategies; 5) **L**ook at the consequences and evaluate the effects of your activity. Although this framework has been adopted and applied in a wide-range of disciplinary fields, it originated in cognitive

This material is based upon work partially supported by the NSF under grants #1623190 and #HRD-1834620. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

psychology [3]. To provide additional support for the learning outcomes, Duke's 7-steps [4] and other execution strategies [5] were used for developing computational solutions to problems. Appropriate learning outcomes in a computer science context were constructed using Bloom's revised taxonomy [6].

Students' competency development in the courses is supported by three distinct teaching methods used in cognitive apprenticeship: (1) modeling, (2) coaching and (3) scaffolding [7]. In modeling, a more advanced problem solver (in our case the instructor) demonstrates paths of problem solving through think-aloud methods to communicate decision stops and considerations of processes. In coaching, the instructor provides feedback for tasks with a particular focus on supporting students' development of new strategies to tackle the problem. In scaffolding, the instructor "controls" those elements of the task that are initially beyond the learner's capacity and provides "scaffolds" to support learning (such as a checklist or the IDEAL framework).

III. REFLECTION

A. Purpose of Reflection

Through a process of reflection, students become aware of their thinking processes, i.e., metacognition, enabling self-assessment of the "why" and "how" of the learning, and what needs to be done as a result. As students reflect on their learning, they gain important assessment information about how they perceive the efficacy of their thinking. Reflection on the IDEAL framework leads to abstraction similarities and differences, reflection on the problem-solving strategy leads to assessment of one's skills level and ability to add what one has learned to his or her toolbox, and reflection on a problem or challenge leads to what has changed in one's understanding of the problem. Reflection further examines what works well in the team and what needs attention.

B. Models for Embedding Reflection

Four types of reflections can be included in the problem-solving course: Mini-Reflection, Problem Reflection, Observation, and Broad Reflection [8]. Each is described next.

Mini-Reflection: At the end of every or every other class period, the instructor prompts a mini-reflection that lasts a maximum of ten minutes. The reflection centers on activities completed during the class period. An example prompt is: Assess the following statements from 1 (disagree) to 10 (agree): I found today's problem easy to understand; I found today's problem easy to solve; I feel comfortable that I could solve a similar problem in the future.

Problem Reflection: The instructor seeks reflection on each student's problem-solving experience for the problems assigned in class, and the reflection is collected as part of classwork. The expected time allotted to the reflection is a maximum of 20 minutes. Example prompts are as follows:

1. Provide specific examples of how your team used IDEAL to understand the problem.
2. How did you decide which were the important elements of the problem? What criteria did you develop for making these decisions?
3. Describe the process used to rephrase the problem.
4. If your team considered multiple approaches or strategies, how did you analyze their appropriateness?

Scoring is based on the quality of reflection for each question: good, average, or poor. A **good reflection** clearly describes all three of the following elements: 1) the process (not the steps); 2) one's experience of applying the problem-solving process to a particular problem; and 3) description of process is unclear. An **average reflection** clearly articulates two of the elements; and a **poor reflection** clearly articulates one or none of the three elements.

Observation. The instructor observes a team working on a problem for approximately ten minutes and provides critical feedback to an individual student or to the team. The instructor can return a rubric with one or more items regarding the process for defining the problem, identifying strategies, proposing solutions, and evaluating potential solutions. Table 1 provides a sample rubric for assessing how the team approached strategy identification. The results are then shared and discussed with the team.

TABLE 1: RUBRIC FOR ASSESSING STRATEGY IDENTIFICATION.

4-CAN SOLVE PROBLEMS	3- IN PROGRESS	2-IN PROGRESS	1- NEEDS TO IMPROVE
IDENTIFY STRATEGIES			
Identifies multiple approaches for solving the problem that apply within a specific context.	Identifies multiple approaches for solving the problem, only some of which apply within a specific context.	Identifies only a single approach for solving the problem that does apply within a specific context.	Identifies one or more approaches for solving the problem that do not apply within a specific context.

Broad Reflection. The instructor seeks a more in-depth reflection about broader competency development at mid-semester and the end of the semester. The reflection is completed during class time and should take approximately ten minutes. The reflection provides the instructor the opportunity to formatively provide students feedback on trends in their development as problem solvers. The following list gives a problem-solving strategy and the attributes of a beginner problem solver versus an informed problem solver. Students rate themselves on a scale of one to ten from where they were at the beginning of the semester and where they are currently. This rubric was adapted from [9]:

- *Understand the problem*: solve problem immediately vs. delay to explore, comprehend, and frame
- *Build knowledge*: pose solutions immediately vs. investigate and research
- *Generate ideas*: adverse to discard, add to, or revise vs. use divergent thinking, brainstorming, and other techniques
- *Weigh options & make decisions*: inability to weigh all options vs. weigh benefits and tradeoffs of ideas

- *Troubleshoot*: no troubleshooting vs. troubleshooting to find appropriate solutions
- *Revise/Iterate*: not revised or are revised in haphazard ways vs. problem solving in a managed/iterative way
- *Reflect on process*: little to no self-monitoring or self-reflection vs. reflective thinking

IV. Evaluation

Post surveys were administered to understand how the problem-solving courses were influencing students' perceptions of their problem-solving skills development [10]. This study complemented the deeper, ethnographic study, described below, as survey items were developed with feedback from the ethnographers following data collection. The survey spans multiple universities—in all, responses were received from seven universities using this curriculum. In this pilot year, surveys were distributed using two different formats— as a link to the online survey distributed in class or following course completion. In both cases, the evaluation team, not the professors, were able to access raw data. During the 2018-2019 school year, 150 problem solving students took the survey. Nearly 60% of the students surveyed identified as Hispanic and approximately 18% as underrepresented by ethnicity/race other than Hispanic (16% African American, 1.5% Native American, <1% Pacific Islander). Twenty two percent of respondents identified as female. A limitation of this survey analysis is the lack of power to address gender and ethnic differences. In future years, as the course expands, this limitation will be addressed.

Participants were asked to describe the problem-solving course as a learning environment, and to consider how the learning environment was or was not supportive of metacognition. This concept was chosen for evaluation because of the reflective nature of the courses developed and the intention of the course to focus on processes (steps taken to understand and solve a problem) over products (correct answers). The highest rated statements related to the course as supportive of explaining solutions (4.47 out of 5), posing questions to better understand problems (4.42), and using a forum for trying creative solutions to problems (4.36).

TABLE 2: RESPONSES TO METACOGNITIVE DEMANDS PROMPTS

Item, Mean (1-5 scale)	
Students are asked to think about how they learn.	4.0
Students are asked to explain how they solve problems.	4.47
Students are asked to think about their difficulties in learning the subject matter.	3.85
Students are asked to pose questions to better understand problems.	4.42
Students are asked to think about how they could become better learners.	3.96
Students are asked to try creative ways to solve problems.	4.36

A. Student-Student Interactions

Changes in the quality of participants' use of specific problem-solving strategies (see Table 3) were, on average, positive—average scores were near 4.0, which corresponded to "better after the course." The largest growth was in "asking questions about the problem to be solved." It appears the scope of the problems were such that there were fewer opportunities to explore more sources of data for problem solving, and for examining cost effectiveness.

TABLE 3: RESPONSE TO PROMPT: "SINCE TAKING THE PROBLEM-SOLVING COURSE"

Item, Mean (1-5 scale, "much worse" to "much better")	
I approach problems using a step-by-step process.	4.09
I ask questions about the problem to be solved.	4.26
I consider multiple variables.	4.15
I consider multiple solutions.	4.16
I reflect on my thinking before designing a solution.	4.11
I think about the problem context (whose problem, the setting of the problem).	4.06
I ask for feedback about my solution while or before I develop it.	4
I consider real-world applications.	4.01
I value solving for cost effectiveness.	3.72
I communicate a problem and a solution in multiple ways.	4.03
I gather data from additional resources when solving a problem.	3.93
I value solving for simplicity.	4.01

TABLE 4: RESPONSE TO PROMPT: BECAUSE OF MY PARTICIPATION IN THE PROBLEM-SOLVING COURSE

Percent of students who "agree" with the item	
I met at least one computer science professional from industry.	54%
I received feedback from (an) industry professional(s).	55%
I learned about the software development interview process.	63%
I practiced skills that will be beneficial to interviewing for competitive computing jobs.	76%
I applied my computer science knowledge to real-world problems.	77%
I am more confident in my problem-solving ability.	86%
I am more committed to getting a job in computer science.	74%

The problem-solving courses were designed to bridge connections between academia and industry (see Table 4). The role of industry partners in course design and interaction and engagement with students was apparent from responses. Because of the dichotomous nature of some of the prompts, the % agree is listed in the table (e.g., students either did or did not receive industry feedback). While just over half had an experience with an industry professional, nearly all are more confident in their problem-solving ability (86%) and over 2/3 felt they practiced skills beneficial to interviewing for competitive computing jobs (76%), which are indicators they may be more prepared to secure competitive jobs upon graduation.

TABLE 5: RESPONSE TO PROMPT: BECAUSE OF MY PARTICIPATION IN THE PROBLEM-SOLVING COURSE

Item, mean, 1=much worse, 5=much better	
I know how to cooperate effectively as a member of a team.	3.98
I have high confidence in my ability to be part of a team.	3.98
I can provide strong support for other members of my team.	3.99
I know how to be a good team member.	3.89
I know about what it takes to help a team accomplish its task.	3.92

Participants described ways in which their course participation shaped their confidence in and performance with teams (see Table 5). Generally, participants responded positively to these items, yet overall these responses were not

as high as the problem-solving skill development items. Instead, they hover just below 4.0 which corresponds to "better."

V. ETHNOGRAPHIC STUDY

A. Methods

As part of a larger study, an ethnographic IRB-approved case study of the PS1 pilot course was conducted in 2017. The pilot lasted six weeks, meeting twice weekly. No prerequisites were required, and undergraduate students at any stage in their CS coursework were allowed to enroll. The CS faculty used riddles to scaffold student learning, and two problem-solving frameworks structured the problem-solving process: IDEAL and Duke's 7-steps. Assignments included: 1) homework reports; 2) exams requiring the rephrasing of problems and reasoning behind a solution; and 3) two interdisciplinary projects with an environmental scientist and an engineer.

B. Data Sources

Data collection for the ethnographic case study included course observations and interviews. Observations focused on social interaction, language, and tool use; interviews focused on the students' experiences and perceptions about problem solving. An abductive approach to data analysis was used to identify emergent patterns in the development and refinement of three salient themes [11]: (a) problem solving as brute force, (b) problem solving as a process of meaning making, and (c) IDEAL framework as pivot in a problem-solving course [12].

C. Findings

The first theme, problem solving as brute force, represents students' initial ideas that problem solving was about finding the fastest solution, or a quick "fix" encompassing an automatic and/or impulsive action. In contrast, problem solving as meaning making speaks to how students' views of problem solving changed in terms of an "aha moment," a shift in "mindset, and/or a "realization" that the "logic" behind the relationship between problem and solution was more important than the fix itself. Here, "logic" refers to a methodical, recursive, reflective and inductive process of making meaning. As a consequence, students revised their view of problem solving from individualistic to a collaborative and inclusive process, placing significant emphasis on communication and social interaction with others in order to consider alternate perspectives about a problem and multiple pathways to diverse solutions [12]. Finally, mediating transformation in problem solving looks at the role the IDEAL framework played in facilitating a change in student thinking and engagement with problem solving. For the students, the IDEAL framework represented a tool that, when introduced into the social practice of problem solving, served to "pivot" [13] students' mindset about problem solving from fixed to a methodical, recursive, reflective and inclusive process with an emphasis on communication and social interaction [12].

VI. SUMMARY

The problem-solving courses created opportunities to learn across academic-year levels and created new student communities, mentorship opportunities, and social connections, supporting retention in the major. The courses incorporated PBL, an established process, and cognitive apprenticeship, including faculty coaching and student self-reflection, that contribute to students' growth as problem solvers. CAHSI faculty continue to refine the course based on shared experiences and evaluation results. Materials for course adoption will be released in early 2020.

ACKNOWLEDGMENTS

The authors acknowledge the contributions of Diego Aguirre, Mayra Bachrach, Tom Carter, Martine Ceberio, Melanie Martin. Nayda Santiago, and Eliana Valenzuela. The authors would also like to extend their appreciation and gratitude to Google (in particular Lorne Needle, Aida Duarte, Corin Hernandez, and Mike Gainer) for their participation in developing the content and dissemination of this effort across CAHSI Alliance institutions.

REFERENCES

[1] C. E. Hmelo-Silver, Problem-based learning: What and how do students learn? *Educational Psychology Review* 16(3), 235-266, 2004.

[2] Coursera, A seven-step approach to solving programming problems, https://www.coursera.org/lecture/duke-programming-web/a-seven-step-approach-to-solving-programming-problems-AEy5M, accessed Mar 2018.

[3] J. Bransford, J. and B. Stein, *The IDEAL Problem Solver: A Guide for Improving Thinking, Learning, and Creativity*, New York: W.H. Freeman, 1984.

[4] H.S. Fogler and S.E. LeBlanc, *Strategies for Creative Problem Solving*, 3rd ed., Upper Saddle River, NJ: Prentice Hall, 2013.

[5] L.W. Anderson and D.R. Krathwohl, *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives,* New York: Longman, 2001.

[6] A. Collins, J.S. Brown, and S.E. Newman, Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics, in L. B. Resnick (Ed.), *Knowing, Learning and Instruction: Essays in Honor of Robert Glaser,* Hillsdale, NJ: Lawrence Erlbaum Associates, 1989, pp. 453–494.

[7] A. Collins, J.S. Brown, and A. Holum. "Cognitive apprenticeship: Making thinking visible," *American Educator, 15*(3), 1991.

[8] D.P. Crismond and R.S. Adams, The informed design teaching and learning matrix, *Journal of Engineering Education*, 101(4), 2012, pp. 738-797.

[9] T. Lam and P. Bengo, *A Comparison of Three Retrospective Self-reporting Methods of Measuring Change in Instructional Practice,* 2003.

[10] J. Saldaña, *The coding manual for qualitative researchers*, 3rd ed., Sage, 2015.

[11] C. Convertino and A. Monarrez. "Problem-solving as sociocultural practice: Creating change in computer science education," poster presented at the Annual Mtng of the *Am Educ Rsch Assoc*, New York, NY, April, 2018.

[12] L.S. Vygotsky, *Mind in Society: The Development of Higher Psychological Processes,* Cambridge, Mass.: Harvard University Press, 1978.