When Does Collaborative Interaction Support Learning of Computational Thinking Among Undergraduate Students

Bushra Chowdhury
Department of Engineering Education
Virginia Tech
bushrac@vt.edu

Dennis Kafura
Department of Computer Science
Virginia Tech
kafura@cs.vt.edu

Aditya Johri
Department of Information Sciences & Technology
George Mason University
johri@gmu.edu

***Abstract*-Collaborative learning has proven to be a productive approach across domains including computer science. Especially for novices, collaboration among learners with diverse experiences, values, and knowledge can be especially effective. However, there is limited research on how collaborative interactions among learners from different undergraduate disciplines manifest during the learning of computational concepts. This qualitative study investigates interaction among learners in small interdisciplinary student groups in an undergraduate computational thinking (CT) class. In analyzing the social interactions of cohort members, the Differentiating Overt Learning Activities (DOLA) framework developed by Chi was used as a starting point. Several themes (e.g. explain-to-other, self-monitor, connect/plan) suggested by Chi's overt activities framework emerged naturally in the observations. However, several categories that were not explicitly stated in the DOLA framework also emerged. These are: asking a shallow question versus asking an elaborate question, socializing, being confused or expressing frustration, self-talk, and struggle. Analysis of interactions between cohort members revealed that most of the group interactions were active in nature. Barriers to collaborative learning were accentuated when peers were novices. Students who lacked knowledge of computational concepts found it hard to contribute to a discussion, hindering learning, whereas students who had better conceptual understanding were constructive in their interactions, elaborated on a topic, made connections with previous problems, and displayed self-awareness about their understanding.**

KEYWORDS-Computational Thinking, Collaboration, Undergraduate Instruction, Qualitative Research, Data Science.

I. Introduction

Computational thinking (CT) has been proposed as a core set of concepts for teaching computing-related skills [1, 2]. According to the Computer Science Teachers Association, it is important for students learning CT to recognize and use computational concepts within their own discipline/interest and realize the variety of problems and scenarios where CT can be used [3]. In this study, we define CT as the ability to formulate or solve a problem using computation elements (e.g. abstractions, iteration, conditional logic, etc.) [4]. Taking a broad view of who should be fluent in CT [1, 5], this study is situated in a CT class for university students majoring in different disciplines (e.g. Political Science, English, Sociology, Animal Science, etc.). To better understand how students from different disciplinary backgrounds learn CT at the undergraduate level this study examined their collaborative learning interactions naturalistically as they worked in small interdisciplinary groups. Chi's DOLA framework is used to better understand the nature of such accounts and how these relate to learning of CT [6].

II. Prior Work

Collaborative learning is an approach that researchers have found to be generally applicable and effective for teaching computer science concepts [7, 8]. Most novice learners of CT, particularly those with a non-science background, can find it difficult to learn how to code. This is because they lack the mental model and domain-specific knowledge of computing [9, 10]. A collaborative learning environment, where students in small groups can freely interact with other group members can create a positive and successful CT learning experience. Interactions in the form of explaining, predicting, arguing, critiquing, evaluating, and defining can help novice CT learners better understand computational concepts [11]. However, just distributing students into groups does not imply that students will collaborate with each other and learn. The social aspects of learning in a collaborative setting are unpredictable, happen naturally, and depend on the behavior of both students and instructors [12]. Some behaviors can lead to increased learning, whereas others may not contribute to learning at all. This study focuses on the social interactions in terms of overt behaviors of group members in order to identify collaborative moments of learning CT. To evaluate such collaborative moments, Chi's Differentiated Overt Learning Activities (DOLA) framework was used [6].

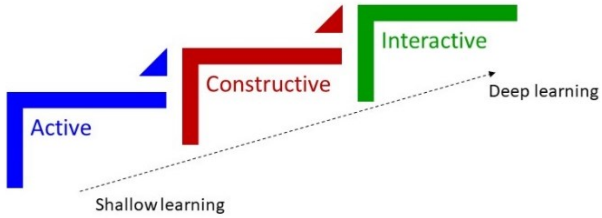

Figure 1 Different types of overt learning activities in the DOLA framework (Chi, 2009)

A.Differentiated Overt Learning Activities (DOLA) Framework

Chi's DOLA framework provides an evidence-based taxonomy to categorize student engagement while learning as active, constructive, or interactive [6]. A student is *active* when s/he is doing something (physically or verbally). An example of an *active* learning would be a student paraphrasing a text or using simulation software to manipulate a computational model. A student is considered to *constructively* engaged when s/he goes beyond what was initially provided. Activities such as self-explanations where a student explains what a text sentence or a solution step means to others in their own words is considered *constructive*. Students are *interactively* engaged when they can explain the subject matter in detail and each partner considers the contribution of the other. An example of an *interactive* activity would be when a student asks an instructor for help and the instructor provides feedback which leads to a more extended dialogue on the issue. Chi's DOLA framework suggests that some behaviors are better in engaging students in deeper learning than others. According to the author, an *interactive* activity involves higher cognitive process than *constructive*, a *constructive* activity is higher cognitively than *active* activity (see Figure 1).

In this study the term 'overt activity', 'social interaction', and 'exhibited behavior' are used interchangeably. These terms refer to an activity that is externally observable (seen or heard) during collaboration. The DOLA framework has previously been used to evaluate student engagement in physics, mathematics, bridge design, evolutionary biology, human circulatory system, introductory materials science and engineering classrooms [6, 13, 14]. The framework has been used largely in quantitative studies to evaluate student interactions between pairs [6, 14, 15]. In this qualitative study, the DOLA framework was used to develop a priori codes to categorize collaborative CT learning behaviors. Using the DOLA framework, this study will illustrate collaborative moments that foster CT learning in novice learners.

III. RESEARCH STUDY

A. Research Objectives and Methods

The goal of this study was to investigate "How do novice CT learners naturally interact with group members while learning CT in a classroom setting?" In this study, social interactions between students are considered as naturally occurring because the students were not obliged to interact with each other. The interactions between students happened spontaneously. This study employs ethnographically-informed data collection methods (e.g., field notes, observation, interview, etc.). Qualitative ethnographic methods have been suggested by Sawyer to be appropriate when the researcher considers social interactions as the mediating mechanism for collaborations contributing to learning [18]. In this study, the collaborative process of learning CT was analyzed by focusing on the social interactions between group members. Thus, according to Sawyer's suggestion, a qualitative investigation seemed best suitable for this study. To better understand the social interactions between group members learning CT in naturalistic settings (in this particular study the naturalist setting is the classroom), focused participant observation was conducted [16]. Observations were conducted during class periods when students interacted with each other (time allotted for open discussion and collaboration). Studies investigating social interaction that used observations as their main data collection strategy usually consider observation data to be the audio or video recording of students working together [17, 18]. Thus, students working in collaboration in the CT class were video and audio recorded. Along with the recordings, one of the researchers also took observation notes during and after observing students' group work, collected student generated computer log data, interviewed the students and examined the final projects of all students. This study was conducted under a protocol approved by the university's board overseeing human subject research.

B. Participants

The study was conducted in a large regional state university in . Southeast U.S. As part of its strategic plan the university has been offering a general education course titled "Introduction to Computational Thinking" to students from all disciplines. In the semester this study was conducted, the class enrolled approximately 45 students and met twice a week for 75 minutes sessions. Within the 75 minutes students worked in groups for 20 minutes. Each computational concept was taught multiple times throughout the course. To facilitate collaboration among students, the class was divided into eight groups. Each group (termed a "cohort") contained students from different majors. Students were assigned to a cohort in the second week of the semester and the formation of the cohort lasted until the end of the semester. While assigning students to cohorts, the course instructor balanced different disciplines and assigned at least two female students to each cohort.

During the face-to-face class, students in a cohort sat together. And performed in-class activities within their own cohorts. In-class activities were not group assignments. Rather, students were expected to solve problems individually, but they were encouraged to interact with their cohort members as needed. To underscore the importance of interaction a cohort "contract" was developed and signed by students of each cohort when the cohort was formed. Students could also collaborate virtually using the course technology, a custom-built, interactive web-based platform with embedded coding activities and real-time, shared text writing (similar to Google Docs). The assumption was that through collaborative discussions, students would gain a better understanding of the computational concepts instead of just learning individually. This collaborative time (20/30 minutes in each class) was analyzed for this study to understand the social interaction among students.

C. Data Collection

Data collection for this study was conducted in one academic term (semester system, approximately 15 weeks). Three student cohorts (cohorts named in this paper as alpha, beta and gamma) working during class time were video recorded during three class sessions. Each video was approximately 20 minutes in duration. On the first and second day of the observations, students were primarily using a block based programming platform. This platform is a visual programming environment that uses composable blocks to make writing code easier. On the third day of observations students were working between the block-based platform and the Spyder (Python) editor, shifting between the two platforms (transferring from block-based coding to text-based coding).

According to the definitions of active, constructive and interactive behaviors provided in the DOLA framework all the class work problems were categorized as Constructive (Table 1). This is because each day's problems required the students to apply what was taught in class on new data sets and they had to gradually integrate different concepts.

Table 1 Type and concept covered in classwork problems during observations

CT Concepts	Type of activity	No. of questions
Conditional Logic\Decision Iteration Dictionary	Constructive	4
Iteration Dictionary Create plots	Constructive	5
Identify Logical Error\Decision Conditional Logic Iteration	Constructive	6

IV. Data Analysis

To analyze the interaction data the steps suggested by Chi for verbal data analysis were followed [19]. First, the video recordings of three days of observations were transcribed and the transcribed data set was used as the sample data set. Second, "episodes" were used as the unit of data for analysis and defined as, "Episodes are characterized as coherent sequences of sentences of a discourse , linguistically marked for beginning and/or end , and further defined in terms of some kind of 'thematic unity'--for instance , in terms of identical participants, time, location or global event or action" [20]. Third, a coding scheme was developed based on Chi's DOLA framework (see Table 2).

Table 2 Operationalized overt behaviors in the context of learning CT based on Chi (2009)

Type of overt activity	Code and definition
Active	*Look/Show*: A student shows another student his/her own code. No elaborate explanation is provided.
	Ask a shallow question: A student asks another student/TA to verify, complete a problem.
	Self-talk: When a student is trying to solve a problem and provides a live commentary of what s/he is doing/says out loud what s/he expects the code to do
Struggling	Both students are trying to solve a problem independently but discuss with each other, both students do not give any concrete feedback
Constructive	*Explain to other*: In response to another student's question a student explains in his/her own words how to solve a problem. This can include the student showing his own code.
	Self-monitor/Reflect: A student expresses what s/he thinks s/he might know or does not know
	Ask an elaborate question: A student asking the question rephrases the question stated in the problem or involves conceptual information or asks a question relating to the process of solving the problem.
	Plan/Connect: A student connects or links a problem to another similar problem. Can plan how to solve the problem.

Fourth, each student in each episode was then characterized as either being active, constructive or interactive based on the adapted framework of Chi's overt activity [6]. Two independent coders coded all data. First the coders identified episodes in all transcripts. Second, the coders applied codes to each student participating in each episode based on their major role in that episode. The coders compared their codes with each other. Disagreement between identifying episodes (18% disagreement) and applied codes (15% disagreement) were discussed between the coders and resolved [21]. Fifth, the number of times a student exhibited a certain behavior was calculated. Graphs presenting the frequency of different overt behaviors for each cohort was also created. This allowed the researcher to gain an overall interpretation of cohort member behavior and to detect patterns across cohorts. Quantifying qualitative data in such a manner has been described by Chi as process where the "researchers rely strictly on the qualitative data, but they quantify the analyses" [19]. This allowed the researcher to compare and confirm subjective interpretation from transcribed data with frequencies of the codes quantitatively.

V. FINDINGS

In analyzing the overt activities of cohort members, the framework developed by Chi was used as a starting point [6]. Several themes suggested by Chi's overt activities framework emerged naturally in the observations. In terms of being active, the category *show/look* was applied to a number of episodes. In terms of being constructive, the category *explain-to-other, self-monitor*, *connect/plan* were observed. However, several categories that were not explicitly stated in DOLA framework emerged. These are: asking a shallow question versus asking an elaborate question, socializing, being confused or expressing frustration, self-talk and struggle. The following paragraphs further describe and justify each category of codes. Figure 2 illustrates the adapted coding framework characterizing students' naturally occurring overt CT learning behavior.

A. *Asking a Shallow versus Elaborate Question*

Chi suggests that "*asking a question*" is a constructive activity [6]. Being constructive requires a learner to state something that is not the verbatim of what is given to the learner. It suggests that the learner is integrating and making inferences about the concept that is not explicitly given [13, 15]. While coding questions the researcher found that not all questions students asked were constructive in nature. At times students asked questions in which they were going beyond what had been given to them in the study material. However, most of the questions could not be categorized under that type. Instead, students simply repeated the questions stated in the classwork problem or asked for direct help without any elaboration of the concept or the problem being solved. This type of asking questions suggested more of an active behavior.

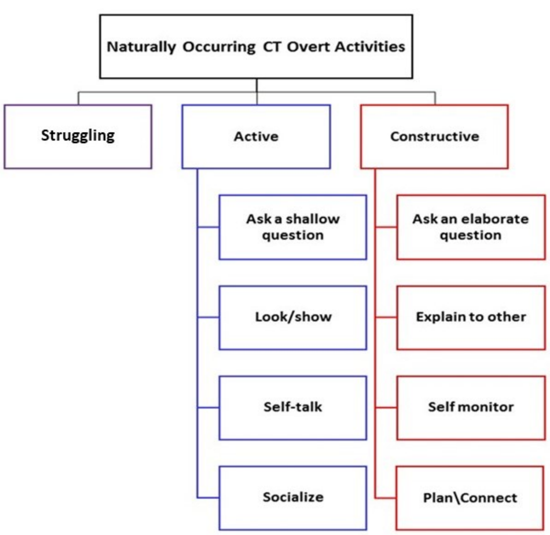

Figure 2 Adapted framework for categorizing naturally occurring CT overt behaviors

Thus, asking questions was divided into two categories: *asking an elaborate question* and *asking a shallow question*. *Asking an elaborate question* was considered constructive and *asking a shallow question* was considered active. In the following quote an International Studies student was asking a shallow question.

International_ Studies_ Major: For number 4 we disregard wind right? (asking a shallow question).

In the following quote a student in Architecture was asking an elaborate question.

Architecture_Major: Did you have your iteration and then your decision inside the iteration?
(asking an elaborate question)

B. *Self-Talk*

While solving a problem, students often gave a spoken commentary of what they were doing. It appeared as if the student was talking with the computer. This type of activity was coded as *self-talk* and was considered an active behavior. The justification for *self-talk* to be active was that the student was speaking out loud what s/he was doing or trying to do. Since, showing or looking at a computer was considered to be active, *self-talk* was also considered to be active. In the following two separate quotes we see an International Studies major engaging in self-talk:

International_Studies_Major: (while working on the computer) You got it. Output. Print that thing. No. Print current temp duh. You've got this. Don't give up. Oh, you! I said you can do it but you obviously can't.

C. *Struggling*

In certain episodes, one or more students were exhibiting a sequence of back-and-forth statements. In such episodes, neither student gave a direct definite response. One student would be trying different options on his/her computer program and explaining half-formed ideas (which were at times also incorrect). Such episodes were coded as *struggling*. Since it was difficult to determine if the student was being active or constructive, it was coded as a separate category. This *struggling* behavior was often noticeable among the novice students, especially among students with non-science background. This could be due to the fact that the syntax and semantics of writing code has to be exact in order for a problem to be solved correctly. If either student in a discussion did not understand the computational concept and how to implement it in the code, the students were coded as *struggling*. In the following excerpt we see two students struggling to extract values from a dictionary in the Python text editor.

Housing_Major: Maybe it's not report.

Internationa_Studies_Major:It's not report. I just changed it to that.

Housing_Major: See, its reporting, reports.

Internationa_Studies_Major: No that doesn't work either. I've already done that.

Housing_Major:Maybe it's uh .. It should be, it's ... uh ... run…

In the above excerpt the two students were going back and forth and trying out different options since neither of them had a clear idea how to solve the problem. For a student to overcome a *struggling* episode the students actively or constructively reached out to others in the cohort or ask help from the TA.

D. Socialize

Students in cohorts with active interactions often talked about other events occurring in their life or inquired about other cohort members social life.

Architecture_Major: I was home for the weekend and got to see X movie. Have you seen it?

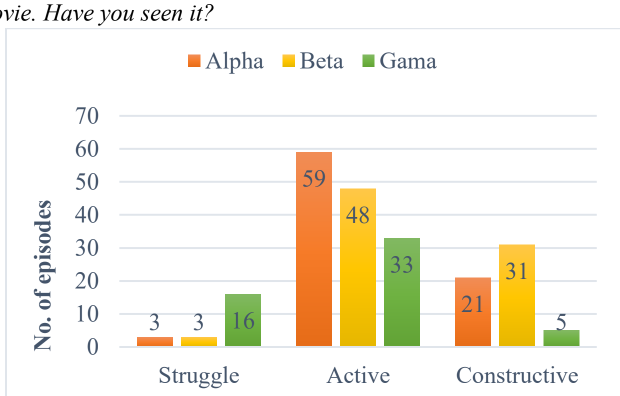

Figure 3 Summary of different types of activities exhibited by cohorts

E. Quantitative results of exhibited overt behavior by different disciplinary students

Figure 3 compares the behaviors of all three cohorts. Active overt behaviors were observed more than constructive and struggle. **Error! Reference source not found.** illustrates cross-cohort comparison of subcategories of overt behavior exhibited by students enrolled in a major in the College of Liberal Arts (8 students), College of Architecture (4 students) and College of Science (1 student). In terms of active behavior students of three colleges asked a shallow question, showed or looked at another student's code, self-talked and socialized. In constructive behavior, students asked elaborate questions, explained in detail how to solve a problem, self-reflects and planed out their solution process. Self-reflection was observed in the English major and a Political Science major only.

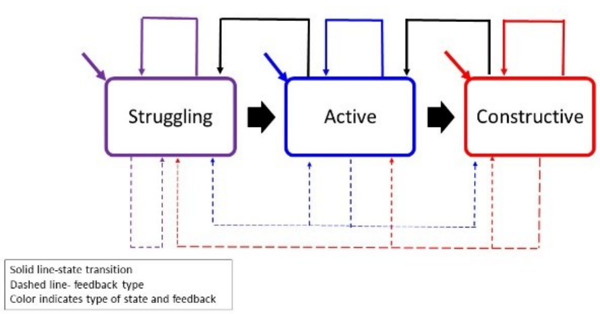

Figure 4 Transition Process

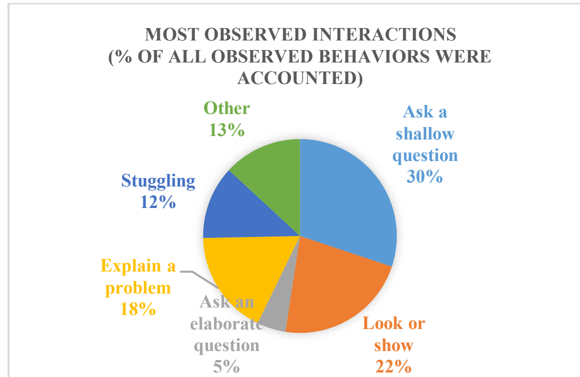

Figure 5 Percentage of all observed behaviors

DISCUSSION

The findings of this study clearly suggest that this particular group of students (mostly majoring in non-science and non-engineering disciplines) were active in nature while communicating with each other. For most of the students, this CT class was the first computer science/programming related course they had taken. The terminology was new, and the way of thinking was new too. While interacting with each other, instead of explicitly saying what the problem was, students would instead show their computer screen, look at another's solved solution – all of which were active in nature, but which indicated a lack of terminology and cognitive ability to frame a verbal description. Getting started with a problem was a common active behavior students exhibited in the CT class.

In multiple episodes, a student was seen to ask another cohort member or the TA what should be the first step for solving a problem. Some cohorts exhibited more constructive behavior than the others. Prior experience in programming and working on a problem for a while allowed students to be constructive. The science major who exhibited more constructive behavior than others had prior programing experience. He was also communicative and concerned about other learners of the cohort and tried to help his peers. Another situation where students were constructive was when they asked a question to the TA after struggling on a problem. In most episodes students who were struggling to solve a problem would try multiple things, ask for help from their peer cohort members and only then ask for help from the TA. Prior to asking help from a TA a student had worked on the problem for a while and had solved certain parts of the problem.

As a result, while asking a question to the TA the student would explain what s/he had done or tried and elaborate on his/her own understanding, exhibiting a constructive behavior. Of all observed behaviors 87% were accounted for by just five behaviors: asking a shallow question, showing/looking at another student's solved problem, asking an elaborate question, explaining to others and struggling on a task. Figure 5 illustrates the percentage of the most observed interactions. As a student started to solve a classwork problem s/he would start off in any of the three states (see Figure 4): struggling, active and constructive.

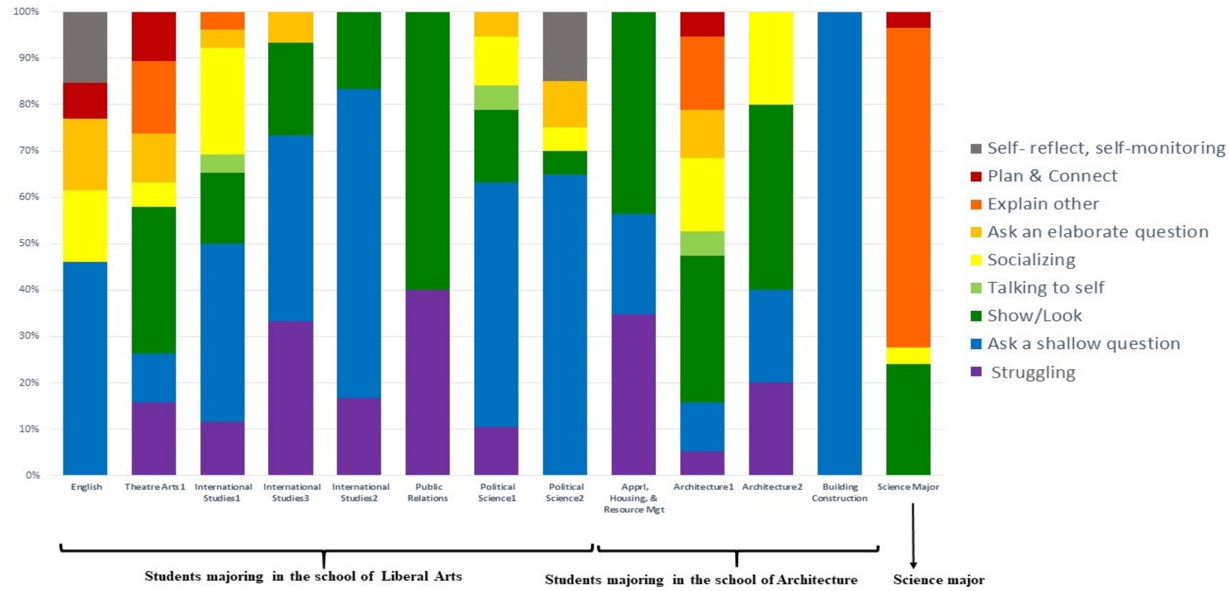

Figure 6 Exhibited overt behavior by different disciplinary student

The transition process (see Figure 4) and its outcomes were influenced by the student's prior knowledge in CT, and the type of support and feedback a student received from cohort members or instructors.

The findings of this research facilitated in formulating the following recommendations for educators regarding the design and assessment of CT related problems:

1. Provide students a collaborative space to learn CT with others
2. Provide students enough time and support within the duration of a class to move through the stages (Figure 4) of *Struggling*, *Active* and excel to become a *Constructive* CT learner. Have teaching assistants assist students working during class time.
3. In forming teams for in-class teamwork it is beneficial to distribute learners with prior programming experience to different cohorts so that each cohort has at least one expert to serve as a means to help other students in the cohort.
4. Since many students are engaged in *struggling* behaviors suggests that they don't have a good understanding of how to get started or what to do next. This implies that students need (1) worked examples giving them a roadmap on the steps in solving a problem, and/or (2) explicit instruction in CT problem solving strategies. These strategies might be in the form of questions to answer.
5. In terms of overt behaviors, there were similar students in different disciplines, suggesting that the different behaviors students display in the class are rooted more in individual differences than in disciplinary choices/aptitudes.

VI. CONCLUSIONS

A collaborative environment can be beneficial for novice CT learners. An environment conducive to collaborative learning allows students to see what other learners are working on, ask for help when needed, and also gain an opportunity to gauge one's own learning. Educators should keep in mind that learners with no prior programming experience and those who are not used to the process of breaking problems into computationally independent sub pieces may find it rather difficult to begin the problem-solving process. Also, the syntax and semantics of writing code can be daunting for some students who are not used the precision involved in writing a piece of code. Thus, novice learners can find it difficult to master the ability to formulate or solve a problem using computation elements. It is important to create a relaxed and supportive learning environment that fosters CT learning and retains novice CT learners. It is imperative for novice CT learners to have adequate time and support within the duration of a class to move through the stages of *Struggling*, *Active* and excel to become a *Constructive* CT learner.

VII. ACKNOWLEDGMENTS

This work is supported in part by U.S. National Science Foundation Awards #142444, 1408674, & 1712129 (PI: Johri) and Award#1624320 (PI: Kafura). 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 funding agencies.

REFERENCES

[1] A. Bundy, "Computational thinking is pervasive," *Journal of Scientific and Practical Computing,* vol. 1, pp. 67-69, 2007.

[2] J. Wing, "Computational thinking," *Communications of the ACM,* vol. 49, pp. 33-35, 2006.

[3] CSTA and ISTE. (2011, second edition). *Computational Thinking teachers resources*. Available: http://csta.acm.org/Curriculum/sub/CurrFiles/472.11CTTeacherResources_2ed-SP-vF.pdf

[4] J. Cuny, L. Snyder, and J. M. Wing. (2010, 3/12/2105). Available: http://www.cs.cmu.edu/~CompThink/

[5] K. Brennan, A. Monroy-Hernández, and M. Resnick, "Making projects, making friends: Online community as catalyst for interactive media creation," *New directions for youth development,* vol. 2010, pp. 75-83, 2010.

[6] M. Chi, "Active-constructive-interactive: A conceptual framework for differentiating learning activities," *Topics in Cognitive Science,* vol. 1, pp. 73-105, 2009.

[7] R. Kavitha and M. I. Ahmed, "Knowledge sharing through pair programming in learning environments: An empirical study," *Education and Information Technologies,* vol. 20, pp. 319-333, 2015.

[8] C. McDowell, L. Werner, H. Bullock, and J. Fernald, "The effects of pair-programming on performance in an introductory programming course," in *ACM SIGCSE Bulletin*, 2002, pp. 38-42.

[9] R. E. Mayer, "The Psychology of How Novices Learn Computer Programming," *ACM Comput. Surv.,* vol. 13, pp. 121-141, 1981.

[10] J. Sweller and P. Chandler, "Why Some Material Is Difficult to Learn," *Cognition and Instruction,* vol. 12, pp. 185-233, 2009.

[11] K. Kreijns, P. A. Kirschner, and W. Jochems, "Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: a review of the research," *Computers in human behavior,* vol. 19, pp. 335-353, 2003.

[12] R. K. Sawyer, "Qualitative Methodologies for Studying Small Groups," in *The international handbook of collaborative learning*, C. E. Hmelo-Silver, Ed., ed, 2013, pp. 126-148.

[13] M. Chi and M. Menekse, "Dialogue Patterns in Peer Collaboration That Promote Learning," *L. B. Resnick, C. Asterhan, & S. N. Clarke (Eds.),* vol. Socializing intelligence through academic talk and dialogue (Chi. 21, pp. 263-274). Washington, DC: AERA, 2015.

[14] M. Menekse, G. S. Stump, S. Krause, and M. T. Chi, "Differentiated overt learning activities for effective instruction in engineering classrooms," *Journal of Engineering Education,* vol. 102, pp. 346-374, 2013.

[15] M. Chi and R. Wylie, "The ICAP framework: Linking cognitive engagement to active learning outcomes," *Educational Psychologist,* vol. 49, pp. 219-243, 2014.

[16] J. P. Spradley and K. Baker, *Participant observation* vol. 195: Holt, Rinehart and Winston New York, 1980.

[17] G. J. Kelly and T. Crawford, "Student's interaction with computer representations: Analysis of discourse in laboratory groups," *Journal of Research in Science Teaching,* vol. 33, pp. 693-707, 1996.

[18] N. M. Webb, M. Ing, N. Kersting, and K. M. Nemer, "Help seeking in cooperative learning groups," *Strategic help seeking: Implications for learning and teaching,* pp. 45-115, 2006.

[19] M. T. Chi, "Quantifying qualitative analyses of verbal data: A practical guide," *The journal of the learning sciences,* vol. 6, pp. 271-315, 1997.

[20] T. A. Van Dijk, "Episodes as units of discourse analysis," *Analyzing discourse: Text and talk,* pp. 177-195, 1982.

[21] Creswell, "Research design: Qualitative, quantitative, and mixed methods approaches," ed: Sage Publications, Incorporated, 2008, p. 4.