Interview Iterations and Improvements for Identifying Intermediate Computer Science Threshold Concepts

Sean Mackay
Department of Engineering Education
University at Buffalo
Buffalo, New York, USA
0000-0002-0358-7594

Brian McSkimming
Engineering Pathways
The University of Oklahoma
Norman, Oklahoma, USA
0000-0001-9363-4974

Adrienne Decker
Department of Engineering Education
University at Buffalo
Buffalo, New York, USA
0000-0002-0822-4813

***Abstract*—WIP Research Paper: Identifying the Threshold Concepts within a discipline illuminates the key concepts or components within the curriculum. Once students have overcome the barrier of learning these concepts, they often will be better able to identify as members of that community, and understanding a threshold concept opens the door to learning additional concepts. Within computer science, there has been much debate over what concepts could potentially be threshold concepts. Meyer and Land originally defined threshold concepts as resulting in an individual being placed into a state of uncertainty or liminality, and successfully traversing this liminal state results in a transformation of the individual with potential feelings of accomplishment. While there has been some work attempting to identify threshold concepts within the first year or beginning stages of programming, little work has considered the intermediate years (years 2 and 3) of university study and what potential threshold concepts exist during this time period. Our goal with this work is to help address this gap that exists by answering the following research question: What do intermediate students identify as being troublesome and/or 'uncomfortable to learn' within their computer science coursework?"**

A first cohort of participants were interviewed in late 2022 and coding began in the first quarter of 2023. The coding of these interviews proved challenging. The students who were interviewed often did not give enough information about a concept for the coders to identify whether the concepts had the key characteristics of threshold concepts. It was considered and accepted that the original interview protocol was not supporting the participants well in eliciting the types of information needed to identify a concept as threshold. The interview protocol was redesigned, and new interviews commenced. The work presented here is a continued discussion of the initial findings and the subsequent change in interview protocol, with the primary improvement being an inclusion of concept mapping. Concept maps, or a graphical representation of the interrelationship of topics and ideas, coupled with an intentional simplification of associated terminology, are expected to reduce cognitive load as participants reflect on their learning experiences. Interviews with the revised protocol including concept maps have been more engaging and productive in identifying potential threshold concepts within the intermediate computer science curricula.

Keywords—Threshold Concepts, Computing Education Research, Computer Science Education, Intermediate Students

I. Introduction

Threshold concepts often represent pivotal concepts within a discipline and determining what these threshold concepts are allows educators and researchers an opportunity to illuminate the concepts students will potentially struggle with the most [1]. The work presented in this work-in-progress research paper highlights our continued efforts towards the development of a qualitative interview protocol aimed at identifying potential threshold concepts within computer science in the middle years of university degree programs. The goal with this protocol has been to allow participants opportunities to discuss their experiences in an effort to identify if concepts that they struggled with possessed the key characteristics of threshold concepts: transformative, initiate liminality, troublesome, integrative, bounding, and irreversible [1]. Once identified, directed efforts can be made to improve the educational experience involving these concepts.

In 2022, an initial interview protocol was utilized to conduct a series of interviews with students in computer science in their second year of study or above [2]. These interviews failed to illuminate sufficient evidence for any of the identified concepts to be characterized as threshold concepts by the research team. However, this setback allowed the research team to revise the protocol for the interviews to better illicit information from the students with regards to the key characteristics of threshold concepts. In this paper, we discuss the initial interview protocol, the issues that were identified, the revised protocol, and some initial findings from the revised protocol. Coded results from this new protocol show promise when compared to the first version and coding efforts of additional interview utilizing the revised protocol are ongoing.

II. Background

Threshold concepts have been discussed in computing education research relatively consistently since the term's introduction in 2003 by Meyer & Land [1]. However, much of the existing literature that has been published by the computing education research community involving threshold concepts has been focused on first year computer science issues. Less research has been focused on identifying threshold concepts faced by intermediate students in their second year or above. Some research has focused on developing concept inventories of threshold concepts within computer science. Eckerdal et al. [13] and Boustedt et al. [3]represent some of the early work aimed at identifying threshold concepts within computer science. Both reports suggest that Object-Oriented

This material is based upon work supported by the National Science Foundation under Grant No. 2044179.

2024 IEEE Frontiers in Education Conference (FIE) | 979-8-3503-5150-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/FIE61694.2024.10893570

Programming met many of the components of a threshold concept.

Rountree et al. published a list of potential threshold concepts that have been proposed for computer science including objects, class declaration, program execution mental models, and program-memory interaction, among others [4]. The authors note that many of these identified threshold concepts are generally focused around troublesome topics experienced by first-year programmers, rather than computer science overall. Other work has proposed pointers, abstraction, object-orientation, and program dynamics as potential threshold concepts, though Mostrom [5] notes that abstraction is not a threshold concept based upon their analysis of transformation biographies and a lack of experiences.

More recent studies of threshold concepts in computer science include Sanders & McCartney's work in 2016 which identified a number of concepts including abstraction, basic programming principles, and class declaration [6]. Other work has tried to identify what threshold concepts exist in computer science by arguing that the acquisition of a particular skill can be seen as a threshold concept in and of itself rather than the concept itself serving as the threshold [7]. Related work by Tedre and Cronje´ [8] further argues the importance of threshold concepts by discussing them in relation to core concepts within the discipline, though they note that there exists a distinction between the two. The authors note the importance of both threshold concepts and core concepts and argue that both play a pivotal role in students' success.

III. The Initial Interview Phase

The initial interview protocol was based upon a study performed by Male and Ballie [9] utilized to inform a complete engineering curriculum revitalization.

Each interview lasted approximately 30 minutes and participants were first provided with a plainly worded definition of threshold concepts and then given a list of "potential threshold concepts" which were identified both from initial research into potential concepts as well as earlier student interviews. The remainder of each interview consisted of a series of questions designed to allow students to reflect on concepts that they self-identified as being potential threshold concepts, encouraging students to discuss topics not already on the list.

The interviews were coded simultaneously by two coders (neither of whom were the interviewers). During the coding sessions, the two coders read the transcript of the interviews at the same time looking to identify points in the interview when an interviewee identified a potential threshold concept. Once a concept was pointed out by an interviewee, the coders then looked at subsequent utterances to determine if the interviewee captured the characteristics of threshold concepts when discussing that concept. The following codes were captured to reflect each of the aspects of a threshold concept: transformative, evidence of liminality, troublesome, evidence of discourse change, irreversible, integrative, bounded.

Despite having identified and considered more than twenty potential concepts recognized by students as being troublesome, very few of those concepts had provided demonstration of transformation, liminality, discourse change, or the rest of the commonly identified characteristics of threshold concepts. The explanations for the lack of identifiable characteristics varied and was full of supposition, ranging from unintended priming of the participants to inexperienced reflection capabilities to recognition of the implicit nature of the ideals.

IV. Protocol Revision Process

After the initial round of interviews had been conducted, coded, and analyzed it became evident to the research team that there were limitations resulting from the design of the original interview protocol. These limitations that precluded moments during the interviews in which utterances indicative of many of the characteristics of a threshold concept could be observed. Consequently, the focus of this research pivoted to learning from these initial interviews and developing a revised interview protocol that would address some of these problems. Several major changes were made to the interview protocol over a series of multiple internal revisions, with one researcher making the change and others reviewing the proposed changes.

A. Threshold Concept Definition

The first major change to the protocol addressed a concern that was felt by the researchers with regards to how the definition for a threshold concept had been worded. The initial definition for a threshold concept had been designed following from the original definition provided by Meyer & Land [1], with an effort to remove jargon from this definition to make it more approachable for participants. The researchers, however, felt that this definition was still too technical in nature and required additional time to simply ensure that the interview participants understood what a threshold concept consists of. Furthermore, it became apparent from some of the interviews that the definition perhaps placed additional extraneous cognitive load on the participants, as a number of interviewees eluded to having difficulties understanding and working through the multitude of parts of the definition.

The definition that we ended up utilizing in this protocol was: *"Threshold concepts are concepts that are recognized as being troublesome; one often finds them challenging and difficult to understand. Understanding these concepts often requires persistence and focused dedication. However, one often will gain great satisfaction when 'it clicks' and you arrive on 'the-other-side' of the hurdle (you have an 'a-ha!' moment). Threshold concepts will fundamentally shift the way you think about the world. This can range from a shift in terms of how you view other topics in your discipline to how you view your entire world. They are also integrative. You will see how different areas of knowledge and experience fit together. In engineering, a threshold concept can be so powerful that complete classes of problems cannot be attempted without first passing through the threshold."* The primary change that was made was to add in language to aid participants in understanding the experience of crossing over the hurdle of learning the concept. Adding language with regards to it "clicking" has thus far seemed to help students understand that portion of the definition better.

B. Adding Concept Maps

One of the more difficult components of a threshold concept to identify from the initial interviews was whether the

participant provided evidence that their identified concept integrated other concepts within the discipline once learned. Another component that was difficult to identify from the original interviews was that of whether or not the concept the participant identified was irreversible in that once they overcame learning the concept it created a permanent shift in their way of thinking. To elicit stronger responses for both components, concept maps were added to the interview protocol.

Concept maps, as discussed by Cañas & Novak [10], serve as a method by which individuals can represent how they conceptualize different concepts as being related to one another. The resulting map is a representation of the relationships between different concepts as the author sees them. Within the concept map, concepts are represented by nodes, with directional vertices between the concepts showing the relationship between concepts. If the author of a concept map, for example, feels that their understanding of recursion directly influenced their understanding of binary trees, they could represent this with two nodes and a directed arc between the nodes. Concept maps have been shown to be effective methods for facilitating learning, originally by Novak & Gowin in chapter 2 of their book [11].

Concept maps have been identified not only as a potential educational tool but also an instrument for measuring students' understanding, allowing an educator or investigator to gain a better understanding of how the student conceptualizes relationships between concepts [12].

To ensure that interview participants were prepared to engage in working with concept maps, the beginning of each interview was designated as time for the interviewer to introduce concept maps. After participants were provided with the reworked definition of a threshold concept, the interview protocol led participants through an example of a concept map of a non-computer science concept, i.e., integrals from mathematics. This allows participants a chance to see concept maps in use prior to having to make their own.

C. Identifying Other Problematic Questions

A third goal with revising the interview protocol was to continue to ensure that leading questions were not present within the protocol. While we did not specifically identify any in the original protocol, a concerted effort was made to ensure that no leading questions were included in the newly revised interview protocol. One question that had be asked in the earlier interview protocol asked participants if they had ever needed to review the concept once they had learned it. Upon reviewing results from the interviews, a number of participants appeared to respond that they in fact did need to, then followed this up by discussing their belief that they might need to in the future or that they would need to when they studied for upcoming exams if the concept was one that they had learned that semester. The original intent of this question was to assess the irreversibility of learning this concept, however after the interviews were analyzed it was deemed that this question was potentially problematic. From the perspective of the researchers, students were thinking about the idea of going back to review the concept as if there was a test coming up and some very specific questions were going to be asked about very specific details. The researchers felt that the student's training around how to study and "review" were being articulated here. The students did not seem to be able to differentiate the need to review for an upcoming examination from needing to review the concept because they no longer understood it. This question was thus removed and not utilized as a primary question in the new interview protocol. Other questions from the original protocol were reused, but had their wording and ordering changed based on other findings from the initial interviews.

V. New Protocol Findings

After the revised interview protocol was finalized, a series of interviews were conducted. Our early findings have suggested that these interviews have been slightly more effective in helping identify traits of potential threshold concepts from these interviews. In this section, we discuss the methodology of these interviews, the data analysis, and the findings that came from these preliminary interviews.

A. Interview Methods

Thus far a total of five interviews have been conducted and coded. Interviews were conducted by two different researchers located at two different universities. Location 1 is a large public research university in the northeastern United States. Location 2 is a large public research institution in the south-central United States. Three interviews discussed in this work in progress were conducted at Location 1, while an additional two interviews were conducted at Location 2. Participants were recruited through in-person advertising in several second through fourth year computer science courses at Location 1, and through a recruitment survey at Location 2. Interviews were conducted via the video conferencing platform Zoom or in person, whichever the participant preferred. Interviews were recorded and transcribed using AI transcription software built into Zoom regardless of interview modality. Transcripts were manually verified for accuracy prior to coding occurring. All transcripts were anonymized, and pseudonyms were given to each participant.

Data was coded using qualitative methods. Codes were predetermined and were based on the definitions of each of the aspects of a threshold concept: evidence of being transformative, evidence of liminality, troublesome, irreversible, integrative, bounded, and discourse change. All five interviews were coded by two researchers, who met several times during this process to ensure that they were consistent in their coding and findings. Interrater reliability prior to discussing the codes was greater than 85%. After discussing the coding of the data and the clarification of some of the codes, Interrater reliability rose to close to 100%.

B. Findings

The interviews conducted with the revised interview protocol suggest that the changes have had positive impact on the results. Some of the participants actively engaged in the process of developing their concept maps, and showed a deep understanding of the concepts they were identifying. However, this was not universal among the interviews, with some participants having to be continually encouraged to work with their concept map. Some participants seemed to struggle with autonomy and ownership of their concept map, with these

participants asking the interviewer if adding something to the concept map would be appropriate or was the "right answer". One interview participant Andrew, in particular, did this after mentioning several concepts he felt related to the potential threshold concept he had identified, often asking "*Okay. So should I write that in the (concept map).*" Despite continual prompting from the interviewer that the concept map was Andrew's and that there were no right answers, he continued to hesitate and seemed to struggle to authentically engage in this portion of the interview.

There were nine unique concepts identified in these interviews. Within the interview's utterances, we were able to code for all seven characteristics of threshold concepts. Evidence of participants having been in a state of liminality is one trait of a threshold concept which still is difficult to identify in these interviews. This was a code that was observed in less than half (4 of 9) of the concepts identified during the interviews. The other particularly problematic code was that of there being evidence of the concept proving to be irreversible in its effect on the participant. We have been able to find evidence that this might be the case in only one third (3 of 9) of the concepts identified from the interviews. We did, however, identify four concepts from the interviews in which we were able to identify at least five of the seven traits of a threshold concept from the contents of the interviews. These concepts were:

- binary (on a hardware level)
- pointers
- searching algorithms
- functions

The new protocol has so far proven to seemingly allow for more promising utterances in alignment with the elements of threshold concepts. Overall, participants seem less confused about the questions they are asked and more readily discuss their experiences. For example, one concept identified by participants in both phases of research has been pointers. Apart from one of the interviews with the original protocol, participants failed to make statements encouraging coding of most of the elements of a threshold concept. In the original interviews, participant A made the following statement with regards to how learning the concept integrated other concepts within computing for them *"Uh it was... It was like crazy like it. Actually, once you realize it, like once I realized it, it made a lot of things easier, like I actually understood what I was doing in other parts. later projects, It was a lot easy, and I realized if I didn't understand this concept, I wouldn't even have able to attempt it, and once you realize it's like euphoria, you you get this like feeling of like relief, and like excitement that you like, actually like, solved what you were looking for, and it's it's like a complete one hundred like you. You go from not knowing it, and then it instantly clicks in your head."* In this utterance, participant A acknowledges that the concept has integrated other things they have learned but they fail to provide specifics. By comparison, when asked to engage with the concept mapping, participant James mentioned specific concepts that he saw as being integrated when asked to engage in concept mapping *"So the main one, like for when I was talking about here, was just data structures like, and why data structures were built, like not built, but like, why they're set up the way they're set up? Why is a linklist like this versus Why is a tree like this? Because if you don't understand, like, pointers, and why they, for one, why you would want a pointer that only go like, like a pointer that only points this way. Because like, for me, when it came to like linked lists, I was like, Well, why would we ever want a single linked list, when a double linked list makes it a little bit easier for you to keep track of everything. But once I understood, like, what they what the, it's gonna be kind, but what the point of pointers were, like, I couldn't think of a better way to say it."* This change in the manner in which participants engaged in these interviews indicates that the inclusion of concept mapping has had a promising effect on how participants have engaged in these interviews.

Functions is the only concept in any of the interviews thus far to have utterances coded to all seven traits of a threshold concept. In fact, all seven traits were expressed by James in his interview. James made statements that seemed to strongly indicate that they feel as though the concept has had an irreversible effect on their view of computing and that their knowledge of the concept could not be undone "*So until I like, fully wrapped my head around them (functions)... Now, when I see programs that don't even have a main, it's like, I'm looking around it doesn't look weird to me or anything. Like if... until I understood that, if someone would have showed me a program and it didn't have a main in it, I'd be like, What is going on here? I'd be like, it'd be confused at what I was reading or looking at.*" In this utterance James directly notes that he can see how learning functions fully has fundamentally changed the way he looks at code and represents one of the first times we were able to find an utterance that strongly pointed to irreversibility. Irreversibility, as noted earlier has been a concept that was very difficult for students to articulate during the first round of interviews. It has not yet been articulated by many of our participants in the revised protocol, but this explanation gives us hope that the trait can be elicited with this new approach.

VI. Future Work

Overall, the changes made to the interview protocol seem to have improved our ability to find the characteristics of threshold concepts in the utterances of our participants. These initial findings provide promising evidence of our eventual ability to identify the threshold concepts in the second and third years of computer science study. Future work aims to conduct more interviews with students with the goal of identifying additional potential threshold concepts. Additionally, while the introduction of the concept map was helpful, there were points in the interviews that the participants did not make as much use of the concept map as we had hoped. We are adding additional guidance in the protocol for the interviewer for keeping the participants focused on and discussing the concept map that they are drawing and engaging with the participants about the contents of their concept map.

References

[1] J. H. F. Meyer and R. Land, "Threshold concepts and troublesome knowledge: Linkages to ways of thinking and practising within the disciplines," ISL10 Improving Student Learning: Theory and Practice Ten Years On, pp. 412–424, Jan. 2003.

[2] B. M. McSkimming, S. Mackay and A. Decker, "Identification of Threshold Concepts for Intermediate Computer Science Students," 2023

IEEE Frontiers in Education Conference (FIE), College Station, TX, USA, 2023, pp. 1-5, doi: 10.1109/FIE58773.2023.10343211.

[3] J. Boustedt et al., "Threshold concepts in computer science: do they exist and are they useful?," SIGCSE Bull., vol. 39, no. 1, pp. 504–508, Mar. 2007, doi: 10.1145/1227504.1227482.

[4] J. Rountree, A. Robins, and N. Rountree, "Elaborating on threshold concepts," Computer Science Education, vol. 23, pp. 265–289, Sep. 2013, doi: 10.1080/08993408.2013.834748.

[5] J. E. Moström et al., "Concrete examples of abstraction as manifested in students' transformative experiences," in Proceedings of the Fourth international Workshop on Computing Education Research, in ICER '08. New York, NY, USA: Association for Computing Machinery, Sep. 2008, pp. 125–136. doi: 10.1145/1404520.1404533.

[6] K. Sanders and R. McCartney, "Threshold concepts in computing: past, present, and future," in Proceedings of the 16th Koli Calling International Conference on Computing Education Research, in Koli Calling '16. New York, NY, USA: Association for Computing Machinery, Nov. 2016, pp. 91–100. doi: 10.1145/2999541.2999546.

[7] K. Sanders et al., "Threshold concepts and threshold skills in computing," in Proceedings of the ninth annual international conference on International computing education research, in ICER '12. New York, NY, USA: Association for Computing Machinery, Sep. 2012, pp. 23–30. doi: 10.1145/2361276.2361283.

[8] M. Tedre, D. Brash, S. Männikkö-Barbutiu, and J. Cronjé, "Towards identification and classification of core and threshold concepts in methodology education in computing," in Proceedings of the 2014 conference on Innovation & technology in computer science education, in ITiCSE '14. New York, NY, USA: Association for Computing Machinery, Jun. 2014, pp. 237–242. doi: 10.1145/2591708.2591758.

[9] Male, S., & Baillie, C. (2012). Engineering Thresholds: An Approach to Curriculum Renewal. Retrieved from Sydney, NSW, Australia: http://ecm.uwa.edu.au/engineeringthresholds

[10] J. D. Novak and A. J. Cañas, 'The Theory Underlying Concept Maps and How to Construct and Use Them'.

[11] J. D. Novak and D. B. Gowin, *Learning How to Learn*. Cambridge University Press, 1984.

[12] S. C. O. Conceição, A. Samuel, and S. M. Yelich Biniecki, 'Using concept mapping as a tool for conducting research: An analysis of three approaches', *Cogent Social Sciences*, vol. 3, no. 1, p. 1404753, Jan. 2017, doi: 10.1080/23311886.2017.1404753.

[13] A. Eckerdal, R. McCartney, J.E. Moström, M. Ratcliffe, K. Sanders, and C. Zander. (2006) Putting threshold concepts into context in computer science education. In Proceedings of the 11th annual SIGCSE conference on Innovation and technology in computer science education (ITICSE '06). Association for Computing Machinery, New York, NY, USA, 103–107. https://doi.org/10.1145/1140124.1140154