Enskilment in the Digital Age: The Interactional Work of Learning to Debug

Virginia J. Flood, University of California, Berkeley, flood@berkeley.edu
David DeLiema, University of California, Berkeley, david.deliema@berkeley.edu
Benedikt W. Harrer, San José State University, benedikt.harrer@sjsu.edu
Dor Abrahamson, University of California, Berkeley, dor@berkeley.edu

Abstract: We present a detailed account of the interactional work between a programming instructor and a middle school student that leads to the resolution of an elusive error in the student's code. By tracing the fine details of how this resolution came to be, we demonstrate how learning to debug in face-to-face interactions resembles a process of enskilment.

One of the most fundamental and difficult aspects of mastering computer programming is learning to *debug*—to detect and resolve hidden errors (*bugs*) in programs (Papert, 1980). Here, we contribute a close examination of a productive debugging encounter between a programming instructor and a fifth-grade student to better understand how face-to-face interactions can support students' search for and resolution of errors.

In the episode we examine, the student is able to locate and resolve a bug in her program without being explicitly told where or what it is. Following Suchman (1987), we approach debugging as an event-driven form of *situated inquiry* that cannot be reduced to pre-specified plans or generalizable procedures. Inspired by ethnomethodological conversation analysis, our fine-grained approach reconstructs the practical interactional work between student and instructor that leads to the error's resolution. We find that the process resembles one of *enskilment* (Ingold, 2000): A newcomer is supported in appreciating and using the affordances of their environment (Figure 1) for *work-relevant perception and action* (Goodwin, 2018).

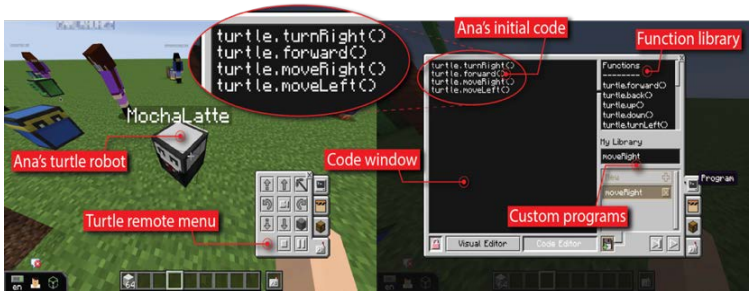

Figure 1. ComputerCraft's integrated development environment (IDE) is a complex, dynamic perceptual field with an array of specialized tools and possibilities for action.

In particular, we find two key interactional mechanisms that contribute to the productive resolution of the bug: (1) The use of *vague references* to occasion the search for the error; and (2) the use of *contracting question agendas* to structure participation in specific forms of perception and action for debugging.

1	((Program runs: Turtle turns right, moves forward, and then stops))	**Adapted Jeffersonian Transcript Conventions:** :: colons for elongated speech; ((blue text)) for action; and [brackets for overlapping action or talk
2	Teo: So, [let's press on- let's press zero to go to your turtle menu	
3	[((Teo points toward screen))	
4	((Ana hits zero, the turtle menu appears and flashes "Program Errored"))	
5	Teo: [And so, you can see something happened, right? The program ran into an error	
6	[((Teo points to turtle menu))	
7	[(("attempt to call nil" appears when Ana mouses over stop button))	

Figure 2. Excerpt 1: Formulating the bug as a witnessable event that can be further specified.

Student Ana writes a program in ComputerCraft to make her turtle robot move one grid square to the right, but the program outcome is not as expected. While Teo, the instructor, could easily spot the problem by watching the program's behavior in the virtual world or inspecting the code in the program window, the nature of the problem is opaque to Ana in both of these domains (Figure 1). Teo could simply tell her from his experience that the error message "attempt to call nil" likely means she is calling a function that has not been defined. However, this would not support Ana in productively engaging in her own search for the bug. Encountering an error without knowing where or what it is and being presented with an ambiguous error message is a pervasive experience in programming. Knowing how to use the IDE to narrow the search to locate the bug is paramount.

Instead, Teo uses evidently-vague (Garfinkel & Sacks, 1970) indexical references ("something happened," and "an error;" Figure 2, E1.5), occasioning further analysis of the issue. Evidently-vague references

are *prospective indexicals* that make presently-unknown, yet-to-be-determined entities phenomenally present in the ongoing interaction. This allows participants to orient to their prospective elaboration and subsequent specification (Goodwin, 2018). Thus, Teo's description creates the need to continue the search.

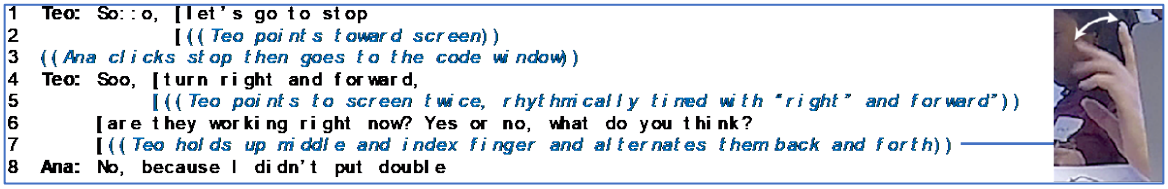

Figure 3. Excerpt 5: Using question agendas to narrow the search: Eliminating properly functioning code.

After a few minutes, Ana has still not discerned which portions of her code are functioning properly. Removing functional lines of code from consideration would significantly simplify the search for the bug. Teo directs Ana to stop the program (Figure 3, E5.1) and calls her attention to two working lines of code: `turtle.turnRight()` and `turtle.forward()` (E5.4), using a rhythmic pointing gesture to highlight each discrete step (E5.5). Then, Teo asks, "Are they working right now?" (E5.6). This polar question projects a restricted, binary yes or no response as the relevant action agenda (Heritage, 2003). Teo *upgrades* this action agenda with the tag question, "Yes or no, what do you think?" (E5.6) By alternating his left index and middle fingers (E5.7), he further amplifies the binary choice for Ana. Goodwin (2000) showed that when gestures appear redundant, they can actually function to *insist* that the recipient address the projected agenda of the talk. By making the required response explicit, Teo makes it difficult for Ana to avoid answering by claiming a misunderstanding of the kind of response Teo is projecting. Designing questions that force recipients to take a binary yes-or-no stand on delicate issues is frequently observed in news interviews. Such questions often take the form of "splits" or "forks," where the interviewee is left to select between two choices the interviewer has given, though neither of them may be appealing (Heritage, 2003). While often seen as an aggressive tactic, Teo's use of his question with its highly constrained action agenda is not hostile or argumentative. Instead, it limits Ana's possibilities for action, creating an opportunity for her to participate in a productive strategy for debugging: Eliminating properly functioning code from the search (Gould, 1975). However, because Ana doesn't realize that these lines work (E5.8), they next move on to examining and testing each statement line by line, until Ana finally finds success.

At the end of this 4-minute episode, Ana was able to fix the bug in her program without ever being explicitly told where it was, what it was, or how to fix it. Ana's path to finding and fixing the bug was not the most efficient or elegant one. However, each strategy Ana was led to participate in along the way was an authentic approach an experienced programmer might choose to locate a bug. Ana's engagement in each approach was emergent from and contingent on her instructor's ongoing efforts to position the digital world in front of her for work-relevant kinds of perception and action. In turn, each of Teo's efforts to position the IDE was finely calibrated to Ana's public exhibits of how she was currently attuned to what she saw in front of her. While taking place in the digital arena of a programming classroom, we find that this process closely resembles the forms of enskilment observed in what are traditionally thought of as more "physical" domains of human practice such as hunting, surgery, cooking, and scientific fieldwork (Goodwin, 2018; Ingold, 2000).

References

Garfinkel, H., & Sacks, H. (1970). On formal structures of practical actions. In J. C. McKinney & E. Tiryakian (Eds.), *Theoretical sociology: Perspectives and developments* (pp. 337–366). New York: APC

Goodwin, C. (2000). Action and embodiment within situated human interaction. *Journal of Pragmatics*, *32*(10), 1489–1522.

Goodwin, C. (2018). *Co-operative action*. Cambridge: Cambridge University Press.

Gould, J. D. (1975). Some psychological evidence on how people debug computer programs. *International Journal of Man-Machine Studies*, *7*(2), 171–182.

Heritage, J. (2003). Designing questions and setting agendas in news interviews. In P. J. Glenn, C. LeBaron, & J. Mandelbaum (Eds.), *Studies in language and social interaction*. Mahwah: Lawrence.

Ingold, T. (2000). *The perception of the environment*. London: Routledge.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books.

Suchman, L. A. (1987). *Plans and situated actions*. New York, NY: Cambridge University Press.

Acknowledgements

This work was supported by NSF AISL #1612660, #1612770, and #1607742.