On Teaching Abstraction in Computer Science: Secondary-School Teachers' Perceptions vs. Practices

Department of Science Teaching

Weizmann Institute of Science

This is an appendix for the paper:

Nakar, L. and Armoni M. (2023). On Teaching Abstraction in Computer Science: Secondary-School Teachers' Perceptions vs. Practices. In *Proceedings of The United Kingdom and Ireland Computing Education Research (UKICER) Conference 2023*. Swansea, Wales.

Additional information regarding the analysis of the class observations in the context of the teachers' actual abstraction-related teaching approach

The classification of the teachers according to their actual teaching approaches in the context of abstraction is mostly based on qualitative analyses of the class observations, in which we considered indications for emphasizing and using abstraction in class. A preliminary thematic content analysis of the transcripts of the observations yielded several types of indications, and these types guided the analyses, as detailed below:

Aspect I: Moving between abstraction levels during the problem-solving process:
We identified in the observations several instances of problem-solving processes (except for T8, for which no instances were found) and chose three of them for each teacher (the most detailed instances, in which the process was explicit and there was a large volume of communication between the students and the teacher). These processes were analyzed from the perspective of abstraction levels throughout the problem-solving process. Specifically, we documented each and every movement between levels of abstraction, resulting in sequences of levels. These sequences were based on an extension of the PGK-hierarchy by some new levels that emerged from the preliminary analysis. The Algorithm level was split into two levels, the second higher than the first:

- Level 3.1 – discussing the solution at the Program level
- Level 3.2 – discussing the solution at the Algorithm level

The Problem level was split into three levels, which do not constitute a linear ordering. Level 4.Introduction.1 is lower than Level 4.Introduction.2, whereas Level 4.Solution is comparable with the 4.Introduction.2 level.

- Level 4.Introduction.1 – introducing the problem at the Program level
- Level 4.Introduction.2 – introducing the problem at the Problem level
- Level 4.Solution – starting the solution process at the Problem level

Aspect II: Abstraction throughout the whole teaching practice:

We started with a qualitative analysis and then quantified some of its results, following Chi (1997). We identified any aspects that reflected abstraction. For example, the level of abstraction of problem descriptions (outside of problem-solving episodes), the use of a top-down problem-solving strategy, and the abstraction level of the criteria used by the teachers for classifying similar problems.

A. Aspects concerning the Algorithm level:
   We identified several aspects that concerned the algorithmic phase of the problem-solving process, a phase that is often skipped (Armoni, 2013). Moving from the Problem level to the Algorithm level involves designing a high-level solution without dealing with implementation details, and hence demonstrates the use of abstraction. Several relevant aspects emerged from the data, such as designing an algorithm and presenting it explicitly, being satisfied with an algorithm alone as a solution (when appropriate) and designing the algorithm with the students (rather than presenting it to them, thus keeping the students engaged with the algorithm design activity and enabling them to master it).
B. Aspects concerning the Program level:
   Several aspects concerning programming emerged from the data. For example, on some occasions the order in which the code was written indicated the use of abstraction (namely, in chunks of blocks of code, which emphasized the meaning of a block of code, rather than sequentially, one line of code after another). Another example is that some of the teachers encouraged the students to write readable code, thus leading the students to think about the meaning of the code, hence rising in the level of abstraction.
C. Abstraction-related didactical aspects:
   Several didactic indications for a teaching approach that aimed at supporting and promoting abstraction emerged from the data. For example, generalizations of any type are manifestations of abstraction, and we found several types of generalizations made throughout the teaching process. For example, general structures of elementary patterns, APs, and guiding rules. Another example for a special type of generalization included incidents in which teachers encouraged their students to find more than one solution to a given problem (this demonstrates the idea that a problem is a generalization of several solutions). Other indications were, for example, an explicit distinction between different levels of abstraction.

Armoni, M. (2013). On teaching abstraction in CS to novices. *Journal of Computers in Mathematics and Science Teaching* 32(3), 265-284.

Chi, M. T. (1997). Quantifying qualitative analyses of verbal data: A practical guide. *The Journal of the Learning Sciences* 6(3), 271-315.