Conceptual Analysis of Analogical Transfer in Common Programming Languages

Rosalind Owen
WestEd, San Francisco, California, United States

Jennifer Houchins
WestEd, Fuquay Varina, North Carolina, United States

Bryan Matlen
WestEd, San Francisco, California, United States

Elysse Caballero
WestEd, San Francisco, California, United States

Kiley McKee
WestEd, San Francisco, California, United States

Yvonne Kao
WestEd, San Francisco, California, United States

Abstract

Analogical transfer is well studied, but much less is known about how students transfer within specific domains. Computer science presents an opportunity to study such transfer, as students often transition from block-based (e.g., Scratch) to text-based (e.g., Python) programming languages. As an early step in understanding programming transfer, we present a conceptual analysis that predicts when students may be aided by analogical supports when transferring from Scratch to Python. We are specifically guided by Structure Mapping theory, which states analogy is a process of aligning objects and relations based on their common structure (Gentner, 1983, 2010). Much research has found that surface similarity influences transfer; thus, we categorized various programming concepts (iteration, booleans, etc.) based on their perceptual similarity. Further, we make predictions about where and how progressive alignment (Kotovsky & Gentner, 1996) can be used to facilitate relational understanding. This analysis sets the stage for future experimental work.

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*