Curing Recursion Aversion

Katherine Gunion and Ulrike Stege
kegunion@uvic.ca, stege@cs.uvic.ca

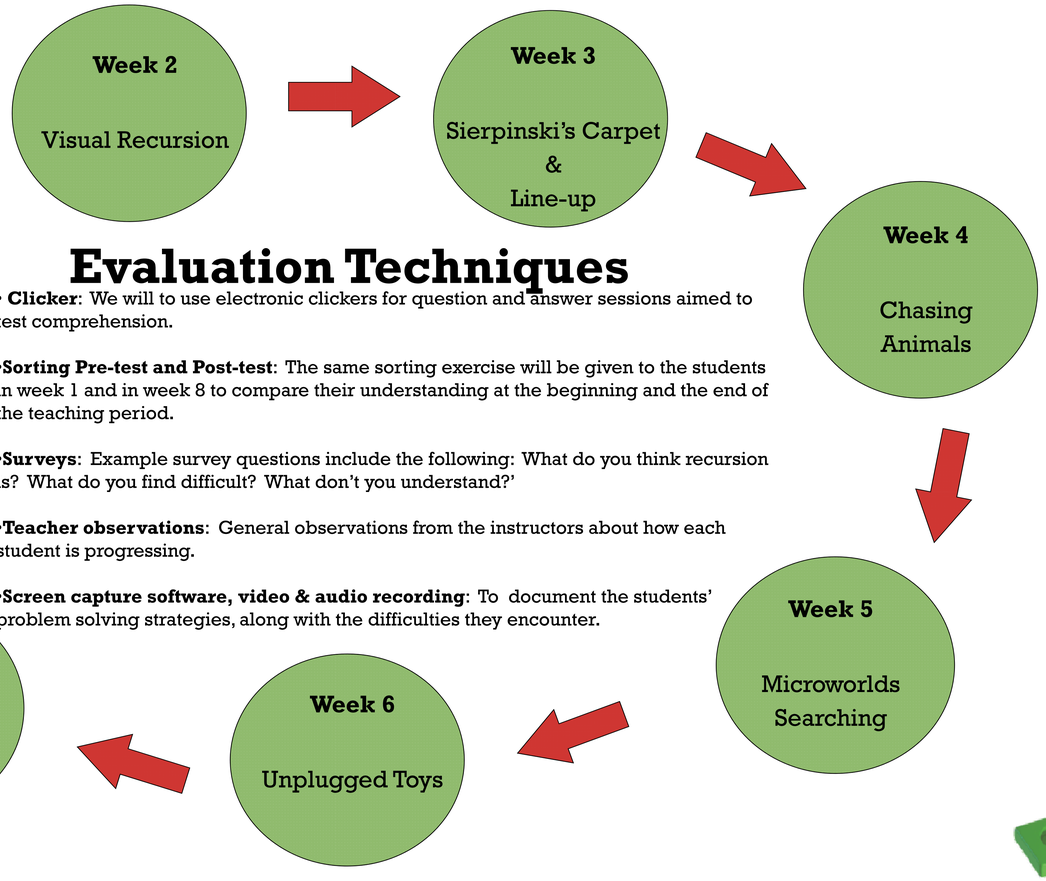
Abstract

At what age are students ready to learn recursion? This question has long been debated in Computer Science education. Anecdotal evidence indicates that many university students struggle with this so-called sophisticated concept. However, we have seen evidence that elementary school students are able to apply recursive solutions to problems given the appropriate abstraction. This poster outlines eight 45-minute sessions that teach recursion with *pre* and *post-tests* to assess comprehension. Also shown are an example survey question, some demonstrative toys and an activity. We will use these tools to demonstrate that recursion can be taught to elementary school students.

Hypothesis

We propose that students as young as 12 will be able to understand recursive concepts given an appropriate level of abstraction. These students will be able to apply recursive solutions to assigned tasks.

Lesson Schedule



Evaluation Techniques

• **Clicker**: We will to use electronic clickers for question and answer sessions aimed to test comprehension.

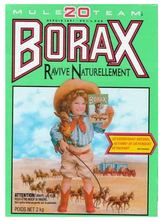
•**Sorting Pre-test and Post-test**: The same sorting exercise will be given to the students in week 1 and in week 8 to compare their understanding at the beginning and the end of the teaching period.

•**Surveys**: Example survey questions include the following: What do you think recursion is? What do you find difficult? What don't you understand?'

•**Teacher observations**: General observations from the instructors about how each student is progressing.

•**Screen capture software, video & audio recording**: To document the students' problem solving strategies, along with the difficulties they encounter.

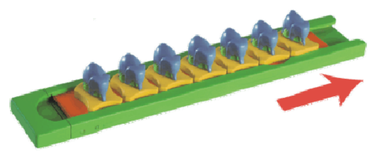
What is recursive about this picture?



A: the horses
B: the box
C: not recursive
D: the lettering

Sierpinski's Carpet

Towers of Hanoi



Elephant Spin-out

Timeline of Related Work

1982 — Kahney and Eisenstadt
• Mental models: copies, loop, odd, null, and syntactic magic [5]

1990 — Bhuiyan, Greer and McCalla
• Recursion is understood in stages: loop, stack, template and model reduction [2]

1999 — Ginat and Shifroni
• Emphasis on concrete level limits understanding
• Emphasis on the declarative abstract improved program formulation [4]

2000 — George
• A graphical tool to help students visualize mental models of recursion[3]

2000 — Veldzquez-Iturbide
• Problem: recursion is taught in imperative languages
• Why: parameter passing and stack must also be covered [8]

2001 — Bergin et al.
• Divide and conquer to teach first year students [1]

2001 — Sooriamurthi
• Emphasis should be placed on WHY to use recursion [6]

2008 — Tung and Chang
• A visual accompaniment to Scheme
• Tool helped trace and solve recursive algorithms [7]

NSERC CRSNG PACIFIC PACIFIQUE
University of Victoria
C R A W
Committee on the Status of Women in Computing Research
THE JADE PROJECT

References

[1]Bergin, J., Kelemen, C., McNally, M., Naps, T., Goldweber, M., Power, C., and Hartley, S. 2001. Non-programming resources for an introduction to CS: a collection of resources for the first courses in computer science. SIGCSE Bull. 33, 2 (Jun. 2001), 89-100.
[2]Bhuiyan, Greer, McCalla. "Mental models of recursion and their use in the SCENT programming advisor". Knowledge Based Computer Systems. (1990), (133-144).
[3]George, C. E. 2000. EROSI—visualising recursion and discovering new errors. SIGCSE Bull. 32, 1 (Mar. 2000), 305-309.
[4]Ginat, D. and Shifroni, E. 1999. Teaching recursion in a procedural environment—how much should we emphasize the computing model?. SIGCSE Bull. 31, 1 (Mar. 1999), 127-131.
[5]Kahney, H. and Eisenstadt, M., "Programmers' mental Models of their Programming Tasks: The Interaction of Real World Knowledge and Programming Knowledge", Proceedings of the Fourth Annual Conference of the Cognitive Science Society, pp.143-145, 1982.
[6]Sooriamurthi, R. 2001. Problems in comprehending recursion and suggested solutions. SIGCSE Bull. 33, 3 (Sep. 2001), 25-28.
[7]Tung, S., Chang, C., Wong, W., and Jehng, J. 2001. Visual representations for recursion. Int. J. Hum.-Comput. Stud. 54, 3 (Mar. 2001), 285-300.
[8]Velázquez-Iturbide, J. Á. 2000. Recursion in gradual steps (is recursion really that difficult?). In Proceedings of the Thirty-First SIGCSE Technical Symposium on Computer Science Education (Austin, Texas, United States, March 07 - 12, 2000). S. Haller, Ed. SIGCSE '00. ACM, New York, NY, 310-314.