The Impact of a Single Lecture on Program Plans in First-Year CS

Francis Castro (WPI)

Shriram Krishnamurthi (Brown University)

Kathi Fisler (Brown University)

Given the Rainfall problem (average non-negative numbers from a sequence), students often produce programs with many errors or non-viable structures

Research shows programmers retrieve and adapt solutions to "similar" problems

How much exposure do students need to potential solution structures?

Study Design

1. Students produce code for 2-3 plan-comp problems

2. Give 50-minute lecture on different plans for those problems

3. Students produce two solutions for different plan-comp problems; preference rank them

A plan-comp problem (e.g., Rainfall)

Solution #1

Solution #2

Preference ranking

Evaluate for structural differences
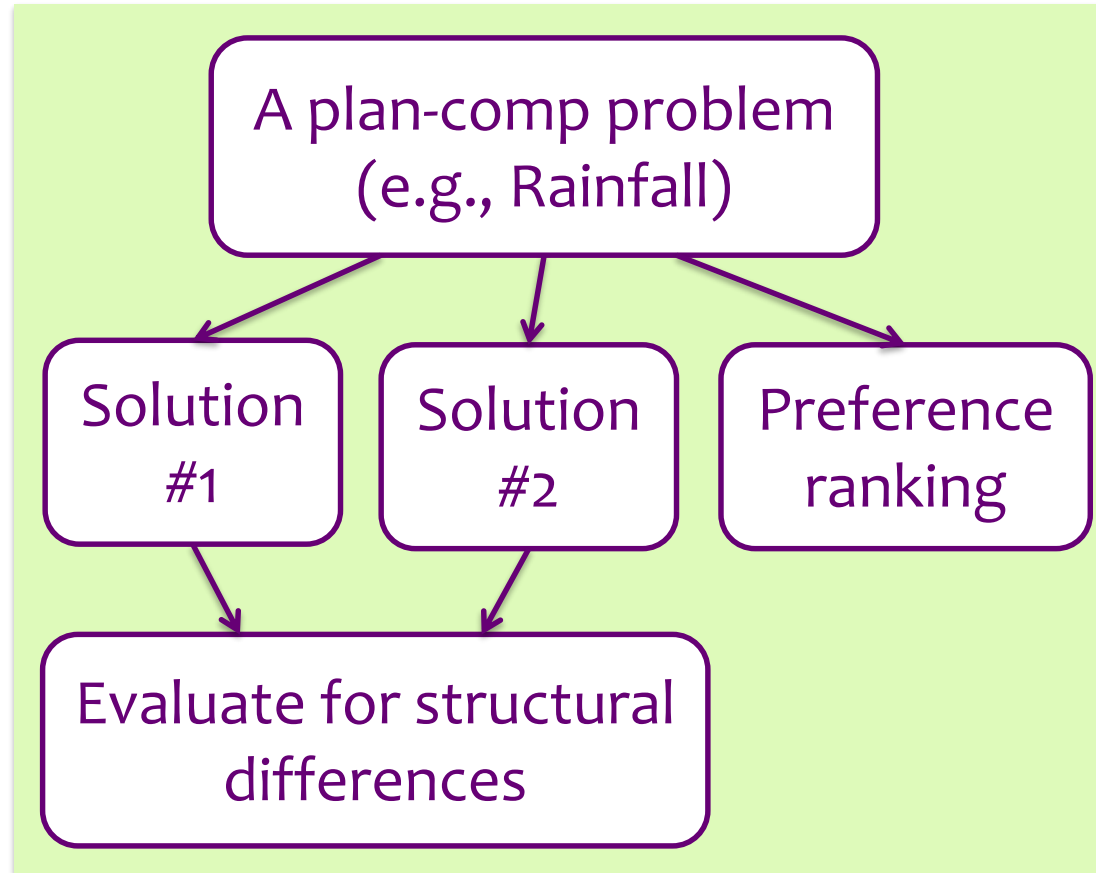
Can students produce multiple different structures after one lecture? Do they prefer different structures than what they produced before the lecture?

Study Design

1. Students produce code for 2-3 plan-comp problems

2. Give 50-minute lecture on different plans for those problems

3. Students produce two solutions for different plan-comp problems; preference rank them

- **CourseA**: "intro" course (functional) for students with significant prior programming (often Java)

- **CourseB-nov**: students in a second course (Java) who were novices in first course (Racket)

- **CourseB-exp**: same course as B-nov, but significant prior programming before first course

Can students produce multiple different structures after one lecture? Do they prefer different structures than what they produced before the lecture?

Pre Lecture Problems

Rainfall: compute average ignoring negatives; truncate at sentinel

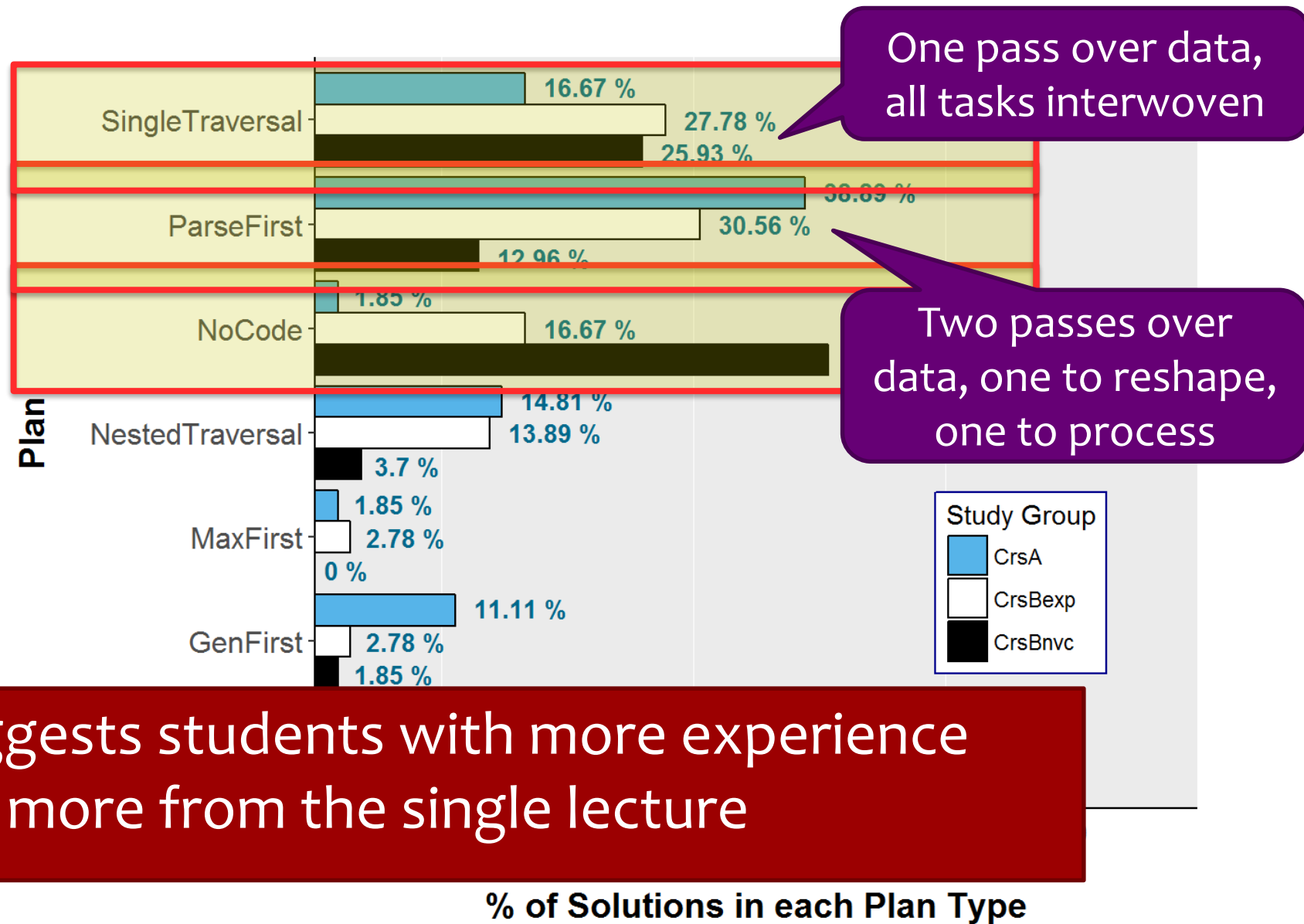*Max Triple Length*: find longest length of 3 consecutive strings in list

Adding Machine: produce list of sums of sublists as separated by zeros; stop at two consecutive zeros

Post Lecture Problems

Data Smoothing: given list of numbers, produce list of averages of each sublist of 3 consecutive elements

Earthquake Monitor: given list interleaved dates and sensor readings, produce list of max reading on each date in given month

Taste of the Findings -- Earthquake

One pass over data, all tasks interwoven

Two passes over data, one to reshape, one to process

Suggests students with more experience get more from the single lecture

% of Solutions in each Plan Type

Taste of the Findings -- Preferences

- **CourseB**: Students preferred diverse solution structures in post, despite all using a single-traversal structure in Rainfall in initial assignment

- **CourseA**: Had a preference ranking question on pre-assessment (on solutions we provided). No clear pattern in structure preferences between pre and post; many students changed preferences

Suggests single lecture does get students thinking about different solution structures

Some Open Questions

- How can we determine whether students can produce multiple solution structures in the pre-assessment? (no vocabulary to describe goal)

- What kind of additional exposure would the novices have needed to perform better? More time discussing planning, more programs written?

- Was exposure to different styles of languages an important factor?