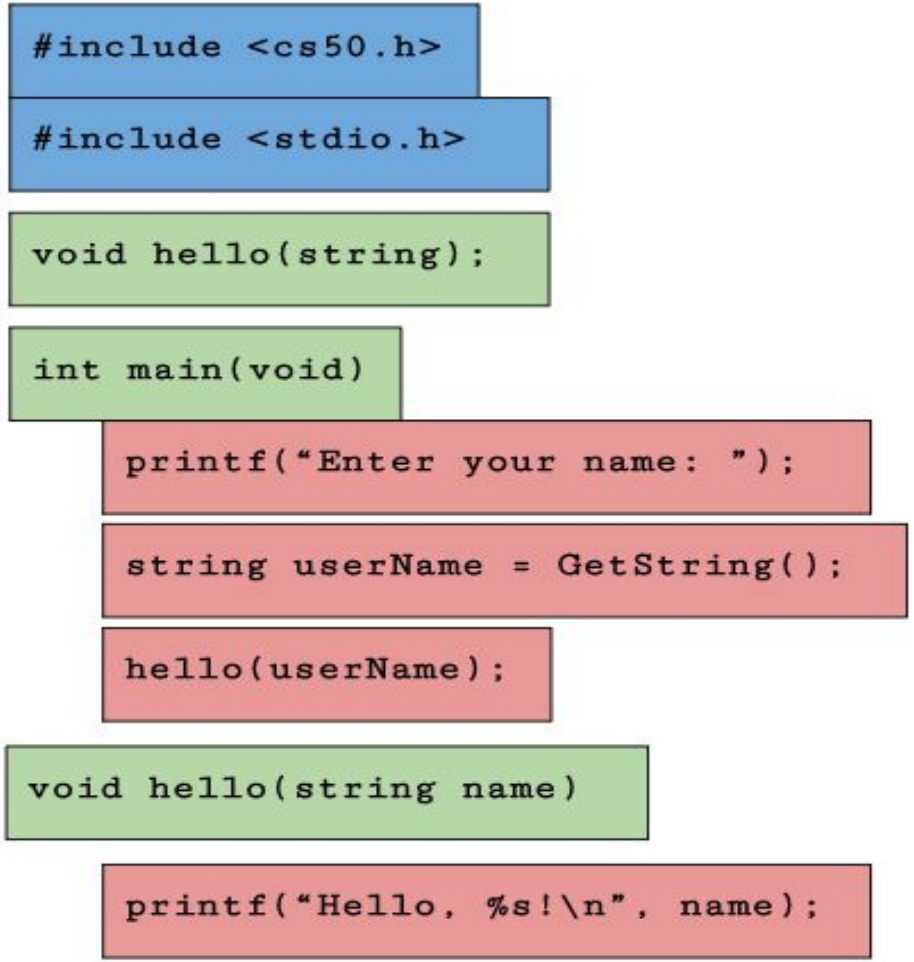
Evaluating the Efficiency and Effectiveness of Adaptive Parsons Problems

Ericson, J. B, Foley, D. J, Rick, J

What is an Adaptive Parsons Problem?



Goal of the Study

Writing Equivalent Code

vs

Adaptive Parsons Problems

vs

Non-adaptive Parsons Problems

Effectiveness
Learning gains

Efficiency
Time spent

Intra-problem vs Inter-problem

If the learner is struggling to solve the current problem, the problem can dynamically be made easier

In the case of this study, the problem could be made easier by pressing a “Help Me” button.

This will provide any number of actions to make the problem easier, such as removing distractor blocks or greying out some incorrect options

The difficulty of the next problem is modified based on the learner’s performance on the previous problem.

1 attempt = made more difficult by unpairing distractors, using all available distractors

4-5 attempts = distractors are shown paired with the correct code blocks

6-7 attempts = 50% of distractors are removed, and remaining 50% are paired with correct code blocks

8+ attempts = all distractors are removed

Example of distractors

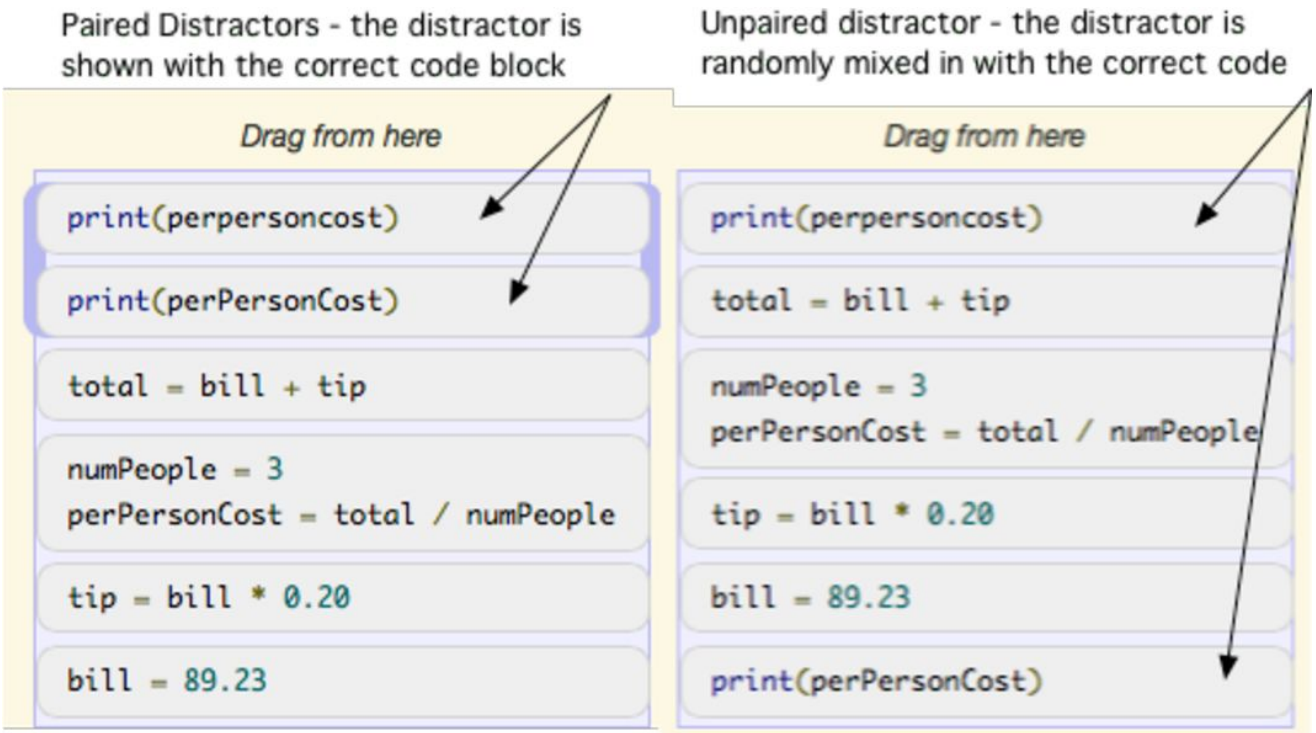


Figure 1

Methodology I

Hypothesis 1

Learners who solve **adaptive and non-adaptive Parsons problems** will finish the instructional problems **significantly faster** than the learners who write code.

Hypothesis 2

Learners who solve **adaptive Parsons problems** will have **similar learning** gains from pretest to immediate posttest as **those who solve non-adaptive Parsons problems and write code**.

Hypothesis 3

Learners who solve **off-task** (not related to the pretest questions) **adaptive Parsons problems** (the control group) will have **lower learning gains than those who solve on-task problems.**

Methodology II

Participants:

- Undergraduate students from a research-intensive university in the US
- All enrolled in one of two sessions for an intro computer science course intended for computing majors
- Both sessions share the same instructor and coursework
- Students were taught to use Python and at the time of the study had been covering files and dictionaries
- Students were told that participating in the entire study could grant them a total of 5 bonus points
- All participants attended sessions at the same time in a closed classroom and were instructed to bring their own material

Methodology III

Session 1:

1) Complete the practice problems which familiarized the students with the online environment and problem types

2) Complete the pretest

- Participants had 15 minutes to complete section 1 which was comprised of 5 multiple choice questions (including tracing code with selection, lists, ranges, and iteration)

- And 10 minutes to complete the remaining 3 sections (fix code, Parsons problem, and write code), answers were saved even if the participant did not complete every question

Session 2:

1) Participants completed the delayed posttest, which was isomorphic to the first posttest

(only variable names and some values were changed, but the structure of the problems was the same)

Methodology IV

Session 1:

- Over the span of 2.5 hours
- Included a consent and demographic survey, a pretest, instructional material, and an immediate posttest
- Instructional material contained four worked-examples. A worked example is a worked out expert solution to a problem

Session 2:

- Lasted 1 hour, a week after the first session
- Included a delayed posttest to measure the retention of material from Session 1

** For the instruction material, students were randomly assigned to one of four practice conditions (groups)
1) solving **on-task adaptive** Parsons problems with **distractors**
2) solving **on-task non-adaptive** Parsons problems with **distractors**
3) **writing the equivalent code** as the Parsons problems
4) a control group that **solved off-task adaptive** Parsons problems with **distractors**

Summary of the Methodology

In Session 1:

- Participants were randomly placed into 1 of 4 groups
- They were given a series of surveys and practice material to familiarize themselves with the system
- Completed a pretest
- Complete their group-specific questions
- Complete a posttest (identical questions)

In Session 2:

- Completion of a delayed posttest with minor changes to variable names and values

Results

Analysis

- 163 students participated in the first session
  - 37 did not answer at least one question or spent less than 30 seconds answering
- Paper only reports on data from the remaining 126 students
- Students were not required to go back for the second session but if they did they earned an additional 2.5 point of extra credit
  - 126 students returned for second session
    - 100 students completed all the questions in both the first and second session, spent at least 30 seconds on each question or got the question correct in under 30 seconds (these students were used to study the retention of the material one week later)

Testing for Efficiency

- The adaptive Parsons (group 1) and non-adaptive Parsons (group 2) had similar mean completion times
- There was a large effect between the non-adaptive Parsons and the write code group and a medium effect between the adaptive Parsons and write code group
- The data fell in a normal distribution
- There was no significant difference in completion time between the adaptive Parsons group and non-adaptive Parsons group
- The time was significantly different between the adaptive Parsons group and the write group as well as the non-adaptive Parsons group and the write group

Table 1: Mean time in seconds and standard deviation for each of the four practice problems by group (condition)

Group	P1 secs (std dev)	P2 secs (std dev)	P3 secs (std dev)	P4 secs (std dev)
1. (*n=32*) A. Parsons	115.65 (50.1)	97.88 (34.3)	191.88 (130.5)	74.63 (23.5)
2. (*n=34*) Parsons	114.29 (56.3)	92.85 (31.0)	190.79 (91.2)	72.94 (26.8)
3. (*n=27*) Write	177.44 (152.0)	118.07 (113.3)	270.48 (152.0)	102 (63.6)
4. (*n=33*) Control	252.24 (100.9)	176.70 (79.6)	178.12 (107.13)	325.06 (160.3)

Testing for Effectiveness

medium effect size for the adaptive Parsons group and a small effect size for the non-adaptive Parsons group

Table 2: Mean score and standard deviation for the pretest and immediate posttest (first posttest) by group

	Group 1 A. Parsons *(n=32)*	Group 2 Parsons *(n=34)*	Group 3 Write *(n=27)*	Group 4 Control *(n=33)*
Pre MC	2.7 (1.5)	3 (1.1)	3.8 (1.4)	3.6 (1.4)
Post MC	3.8 (1.1)	3.4 (.17)	4.3 (1.3)	4.2 (1.2)
Pre Fix	8.1 (1.6)	8.9 (2.0)	9.0 (1.6)	8.8 (1.8)
Post Fix	9.2 (1.8)	9.6 (2.0)	9.8 (1.8)	8.8 (2.1)
Pre Order	7.3 (3.3)	8.6 (3.0)	7.7 (3.4)	7.4 (3.6)
Post Order	8.5 (3.0)	9.5 (1.8)	8.0 (3.3)	7.9 (3.5)
Pre Write	8.6 (2.3)	9.3 (1.3)	9.0 (1.7)	9.2 (1.2)
Post Write	9.3 (1.3)	9.4 (1.0)	9.0 (1.9)	9.2 (1.4)

Testing for Effectiveness

statistically significant change from pretest to the immediate posttest using a multivariate analysis of variance

Bonferroni post-hoc test does not indicate a statistically significant difference by condition

Table 3: Mean score and standard deviation for the pretest, immediate posttest (first posttest), and delayed posttest (2nd posttest) by group

	Group 1 A. Parsons *(n=27)*	Group 2 Parsons *(n=30)*	Group 3 Write *(n=19)*	Group 4 Control *(n=24)*
Pre MC	2.7 (1.5)	2.9 (1.1)	3.8 (1.3)	3.8 (1.5)
1st Post	3.9 (1.0)	3.7 (1.6)	4.2 (1.5)	4.3 (1.3)
2nd Post	3.8 (1.1)	3.7 (1.2)	4.3 (1.1)	3.8 (1.2)
Pre Fix	8.1 (1.6)	8.9 (2.0)	8.9 (1.6)	8.6 (1.9)
1st Post	9.3 (1.7)	9.6 (2.0)	9.4 (2.0)	8.8 (2.2)
2nd Post	9.1 (1.8)	9.7 (1.9)	9.5 (1.8)	9.3 (1.6)
Pre Order	7.7 (3.1)	8.1 (3.3)	6.8 (3.7)	7.2 (3.8)
1st Post	8.2 (3.1)	9.4 (2.0)	7.1 (3.7)	7.4 (3.7)
2nd Post	8.7 (2.4)	9.7 (1.4)	9.2 (2.0)	8.3 (2.9)
Pre Write	8.9 (1.9)	9.4 (1.4)	8.7 (2.0)	9.1 (1.3)
1st Post	9.4 (1.2)	9.5 (1.1)	8.7 (2.2)	9.0 (1.5)
2nd Post	9.3 (1.1)	9.7 (1.0)	9.0 (2.1)	9.3 (1.3)

Testing for Effectiveness

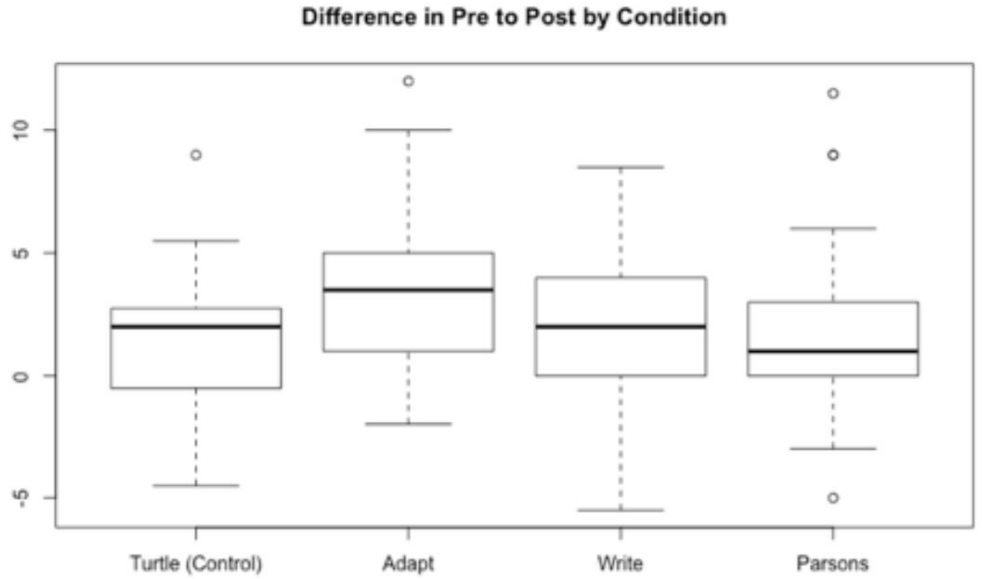


Figure 10: Results from a Mann-Whitney U test comparing the pretest score to the immediate posttest score by condition

Analysis of Demographic Information

- First session total of 126 students
  - 73 (58%) male
  - 51 (40%) female
  - 2 did not answer
- Older students did worse than younger
- Males performed better than females

Biases

Limitations that were mentioned:

- Some learning gains were from answering the same or similar problems with correctness feedback
- The relative effectiveness of intro-problem and inter-problem adaptation is unknown
- Results are only from undergraduates from one university

- Undercoverage
- Volunteer bias
    - Weren’t forced to participate
    - Those who did participate were incentivised

Conclusion

Hypothesis 1	Hypothesis 2	Hypothesis 3
Learners who solve **adaptive and non-adaptive Parsons problems** will finish the instructional problems **significantly faster** than the learners who write code. Supported	Learners who solve **adaptive Parsons problems** will have **similar learning** gains from pretest to immediate posttest as **those who solve non-adaptive Parsons problems and write code**. Supported	Learners who solve **off-task** (not related to the pretest questions) **adaptive Parsons problems** (the control group) will have **lower learning gains than those who solve on-task problems.** **Not fully supported**

Conclusion

Gaps in the Research, What Happens Now?

Solving either adaptive Parsons problems or non-adaptive Parsons problems is a more efficient and an equally effective way to practice coding.