Final Report
Parson's Programming Puzzles: Optimizing Efficiency and Investigating the Effects of Feedback

Further research on Social Addictive Gameful Engineering (SAGE) design and computational thinking (CT)

Spring 2021

Alexander Liebeskind

Overview

1. Related Research
2. Study Purpose
3. Methods
4. Results
    a. Cognitive Load
    b. Performance
    c. Efficiency
    d. Motivation
5. Discussion
6. Further Work

Related Research

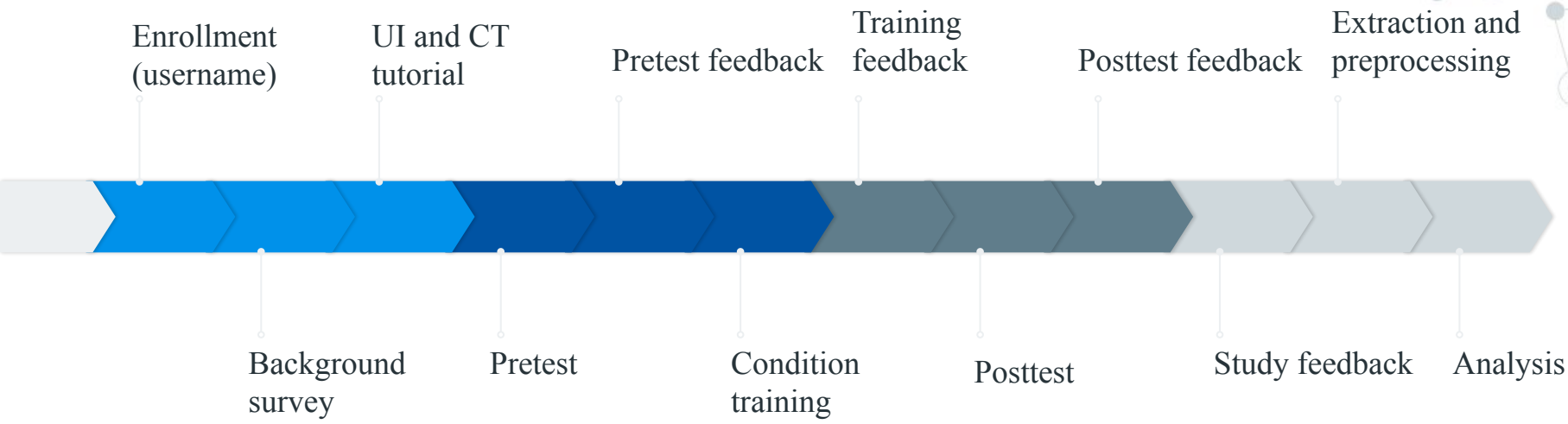
1. Integrating Parsons Programming Puzzles with Scratch
2. Parson's Programming Puzzles: A Fun and Effective Learning Tool for First Programming Courses
3. Lessons Learned from Available Parsons Puzzles Software
4. Measuring Cognitive Load in Introductory CS: Adaptation of an Instrument
5. Instructional Efficiency: Revisiting the Original Construct in Educational Research

Study Purpose

Number	Type	Name	Game Type
1	PPP+f	Let's go sailing! (A)	parsons
2	PPPd+f	Let's go sailing! (B)	parsons
3	ScratchIE+f	Let's go sailing! (C)	parsons (no palette)
4	ScratchIEd+f	Let's go sailing! (D)	parsons (no palette)
5	PPP-f	Let's go sailing! (A2)	parsons (no feedback)
6	PPPd-f	Let's go sailing! (B2)	parsons (no feedback)
7	ScratchIE-f	Let's go sailing! (C2)	parsons (no feedback no palette)
8	ScratchIEd-f	Let's go sailing! (D2)	parsons (no feedback no palette)
9	PPP+f	Your very first recipes! (AC)	parsons

1. Fs1/SAGE validation
2. Cognitive Load, Efficiency, Performance, Motivation
3. Demographics

Methods


Enrollment (username)
UI and CT tutorial
Pretest feedback
Training feedback
Posttest feedback
Extraction and preprocessing
Background survey
Pretest
Condition training
Posttest
Study feedback
Analysis

Results: Cognitive Load

1. After Pretest (Survey 03)
2. After Puzzle (Survey 04)
3. After Posttest (Survey 06)

Test Statistics[a,b]

	Intrinsic Load	Germaine Load	Overall Cognitive Load	Extraneous Load
Kruskal-Wallis H	24.554	3.845	24.818	20.305
df	7	7	7	7
Asymp. Sig.	.001	.797	.001	.005

Results: Performance

Test Statistics[a]

	PRE - POST
Z	-2.735[b]
Asymp. Sig. (2-tailed)	.006

Figure 6.2: Sample output for Wilcoxon Signed Ranks Test indicating a significant difference in transfer performance for groups 1-8.

Test Statistics[a]

	PRE - POST
Z	-1.029[b]
Asymp. Sig. (2-tailed)	.303

Figure 6.3: Sample output for Wilcoxon Signed Ranks Test indicating no significant difference in transfer performance for group 9.

Results: Efficiency

$$E_{instructional} = \frac{Z_{P_{test}} - Z_{E_{learning}}}{\sqrt{2}} \qquad E_{performance} = \frac{Z_{P_{test}} - Z_{E_{test}}}{\sqrt{2}}$$

Test Statistics[a,b]

	Performance Efficiency (time based)	Performance Efficiency (CL based)	Instructional Efficiency (time based)	Instructional Efficiency (CL based)
Kruskal-Wallis H	4.721	7.264	40.215	14.808
df	7	7	7	7
Asymp. Sig.	.694	.402	.000	.039

Figure 6.4: Sample Kruskal Wallis H test efficiency output.

Results: Motivation

1. TEQ Results
2. Individual quantifiers
3. Expanding fs1 results to further conditions

Programming is...	PPP	PPP-distractor	limited-constraint-feedback
Positive Shifts			
something I've wanted to learn	M=0.19, SD=1.40	M=0.27, SD=1.31	M=0, SD=1.19
fun	**M=0.74, SD=1.67***	M=0.40, SD=1.74	M=0.36, SD=1.43
enjoyable	**M=0.90, SD=1.83***	M=-0.05, SD=1.68	**M=0.68, SD=1.76***
important to know	M=0.25, SD=1.48	M=-0.05, SD=1.17	M=0.09, SD=1.19
easy to start	**M=1.35, SD=2.29***	**M=0.68, SD=1.13***	M=0.45, SD=1.71
something that takes practice	M=0.065, SD=1.09	M=0.05, SD=1.29	M=-0.32, SD=1.17
Negative Shifts			
too difficult to understand	**M-1.48=, SD=2.03****	M=-0.77, SD=1.77	M=-0.64, SD=1.89
boring	M=-0.41, SD=1.6	M=-0.32, SD=1.17	M=-0.54, SD=1.90
a foreign concept	**M=-1.13, SD=1.83***	M=-0.27, SD=1.55	M=0, SD=2.07
too time consuming	M=-0.35, SD=2.09	M=-0.09, SD=1.27	M=-0.09, SD=2.44

Discussion

1. Significance of performance and instructional efficiency findings
2. Efficiency support for related work
3. Informing fs3 study design
    a. Sample size, pipeline, conditions
4. Key points for SIGSCE submission

Further Work

1. Fs1 revisions

2. Fs3 design

3. Fs2 writeup (SIGSCE technical symposium)

4. Additional analysis
    a. Demographics
    b. Grouped comparisons (training element, puzzle type, etc.)

References

[1] SDT. Self-determination Theory. https://selfdeterminationtheory.org/intrinsic-motivation-inventory/.

[2] B. B. Morrison, e. a. (2014). Measuring cognitive load in introductory cs: adaptation of an instrument.

[3] Briana B. Morrison, Brian Dorn, M. G. (2014). Measuring cognitive load in introductory cs: Adaptation of an instrument. ICER ’14: Proceedings of the tenth annual conference on International computing education research.

[4] Dale Parsons, P. H. (2006). Parson’s programming puzzles: A fun and effective learning tool for first programming courses. ACE ’06: Proceedings of the 8th Australasian Conference on Computing Education - Volume 52.

[5] Dmytro Vitel, Bari A.T.M. Golam, A. G. (2019). Lessons learned from available parsons puzzles software.

[6] Fred G. W. C. Paas, J. J. G. V. M. (1993). The efficiency of instructional conditions: An approach to combine mental effort and performance measures.

[7] Jeff Bender (2021). Spring 2021 field study goals.

[8] Jeff Bender, Bingpu Zhao, L. M. G. K. (2020). Integrating parsons programming puzzles with scratch.

[9] P. Charters, e. a. (2014). Challenging stereotypes and changing attitudes: the effect of a brief programming encounter on adults’ attitudes toward programming.

[10] Tamara Van Gog, F. P. (2008). Instructional efficiency: Revisiting the original construct in educational research.