Designing Educational CS Games to Support Equitable Access

Stacey A. Koornneef • Jeremy S. Bradbury • Michael A. Miljanovic
Faculty of Science • Ontario Tech University • Oshawa, ON, Canada
stacey.koornneef@ontariotechu.net, jeremy.bradbury@ontariotechu.ca, michael.miljanovic@ontariotechu.ca

Motivation

- Educational, or serious, games are becoming increasingly popular
- We believe that these games and tools should be designed to provide equitable access to students with and without access to digital devices
- Games and tools can be distinguished between the way the interaction mode for creating programs and observing the solution of what's created, but each has a cost associated with it, making some tools more cost accessible than others

RESEARCH GOAL:

Create a computational thinking game for K-5 students that is cost accessible and designed to be cooperative and engaging.

RESEARCH QUESTIONS:

RQ1: *In educational programming games, how does a tangible vs hybrid interface support learners' abilities to make and fix mistakes?*

RQ2: *In educational programming games, how does a tangible vs hybrid interface affect collaboration and/or engagement?*

Background

- **Educational Games** are used to teach a user a skill or technique through the use of a game, also known as serious games. A common example of this in computer science and computational thinking is Code.org [1].
- **Block-Based Programming** is a method of using blocks of small segments of code to create a program. This allows users to learn to code while avoiding issues with syntax and programming language [2].

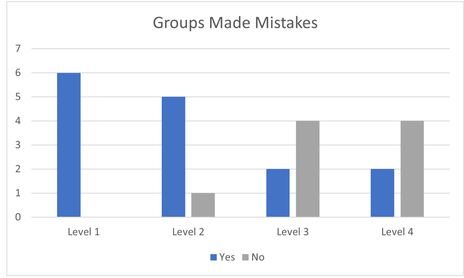
Tangible vs Digital

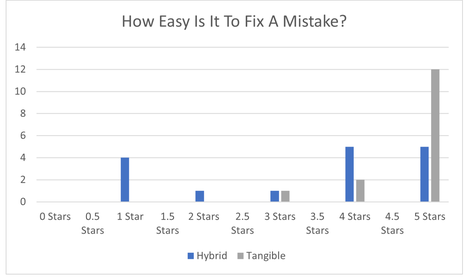
- Not everyone has access to digital devices to learn computer science
- Students may not be able to practice outside of the classroom
- Games that are completely digital require a digital device and sometimes internet and access, which can become expensive
- There are inexpensive alternatives to digital games

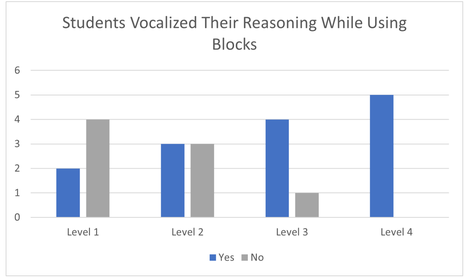
		CREATING CODE	
		DIGITAL	TANGIBLE
OBSERVING SOLUTION	DIGITAL	$$	$$
	TANGIBLE	$$$	$...$$$

Results

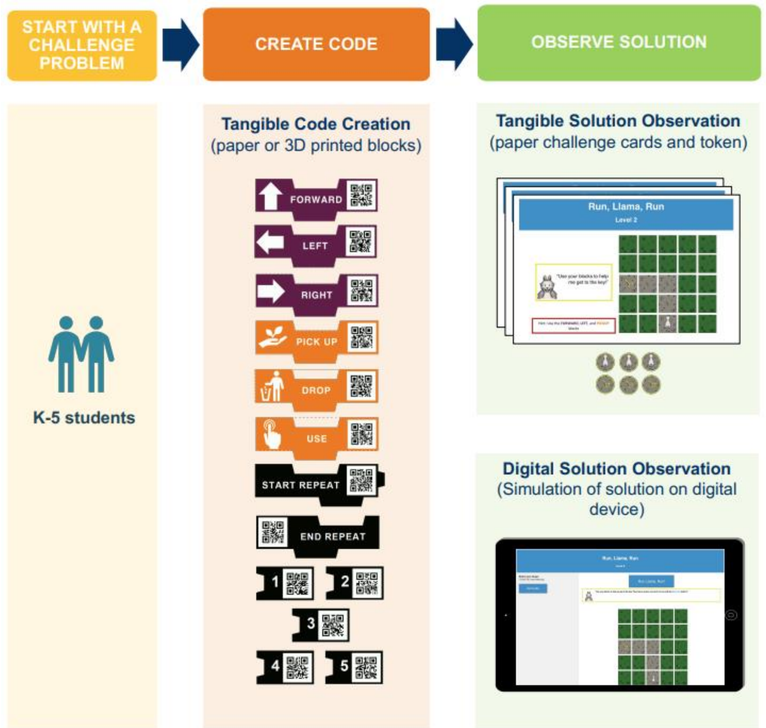
- Students made less mistakes during the tangible version of the game
- Students found it easier to find mistakes and get the llama to the apple in the tangible version
- Students enjoyed playing together more with the tangible version than the hybrid version







Run, Llama, Run



Conclusions & Future Work

- The tangible interface decreased the amount of mistakes made, but removed some of the rules that are common in similar educational games for computational thinking
- Students were more likely to speak with their group members about the game when using the tangible version and it was less likely to have one student take over the game
- Future work includes creating a digital version and a digital coding, tangible solution version to observe the differences between all version, as well as further studies to observe how effective the game would be for learning

References

[1] F. Kalelioğlu. 2015. A new way of teaching programming skills to K-12 students: Code.org. Computers in Human Behavior 52 (2015), 200–210.

[2] D. Weintrop. 2019. Block-based programming in computer science education. Communications of the ACM, 22-25.

FUNDING PROVIDED BY: NSERC CRSNG