Create a Digital Automata - Project One

Learning Goals:

- **Use metacognition skills to examine your own learning and thought processes while learning to use CoSpaces.**

Whether or not you have programmed before, learning something completely new triggers an opportunity for you to examine how you approach a new problem or situation. Do you read instructions? Dive right in? How do you deal with a roadblock or frustration? Do you enjoy success? At what point do you know you have 'mastered the task?'

- **Use basic coding commands to create a digital automata and have a shared programming experience as a course reference.**

In this course we will be referencing computer programming and algorithmic tasks. This little project will give us a shared programming experience. If you are new to coding, all the basic ideas are here. If you are an expert, try to see how these simple blocks reflect advanced concepts.

- **Use a digital automata to reinforce the themes of Module One, "Mind and Machine."**

As you create your digital automata, consider how it is the same and different from mechanical automata. What features make your creation more 'real?' Is a mechanical automata more wondrous? How can we use this example to explore the idea, "mind is to brain and software is to hardware?"

Deliverables:

- A Cospaces activity created individually or with pairs.
- Mini Project Narrative - submitted and graded in Gradescope.

Creating a CoSpace Digital Automata

Login to https://cospaces.io/edu/ with your colorado.edu account.
Enter class code 2RP6G - if needed we may open a second Copsace class.
If you'd like to work with a partner to make a pair of interactive automata - contact Prof. Stade.

Background
If you have never programmed before, go to https://blockly.games/ to try some simple code games first.

Mechanical Automata: Watch examples of mechanical automata on youtube:

https://www.youtube.com/watch?v=YAg66jrvpHA (first six minutes)
https://www.youtube.com/watch?v=DgIDStgaybc

https://www.youtube.com/watch?v=L3Die7PfKvo
https://www.youtube.com/watch?v=-OJ1Yc2SwAs

Watch the CoSpace video tutorial.

Now Create a Digital Automata in your Cospace assignment

Your automata should:

- Be an original creature in Cospaces (i.e. not the from the demo) in the spirit of mechanical automata.
- Create the action(s) of a living creature - make it seem 'alive' using an 'action' and speech.
- Include dialog
- Include movement.
- Option - with a partner, create automata that interact with each other.

Submit the narrative as PDF in Gradescope.

- *For 1 - 4 the content **must fit exactly on the pages given.***
- *Include these first 2 pages as is.*
- *Add as many pages as needed at the end for screenshots.*
- *Don't forget to include your name.*

Digital Automata Project Narrative - Your Name here: Alexander Hawkins

1. Describe your Digital Automata - What does it do?

My automata is a representation of a drifting course. The course is set upon sand dunes and a field of grass in the center of the map, and various animals and people in the drifting course. The automata begins with a character named Papa Drift King who welcomes the driver and explains how the course works. Once Papa Drift King is done talking, the driver can click on him and the drifting will begin. The driver goes around the course and finishes all the way through with a cheerful celebration at the end.

2. In 3-5 paragraphs, on this page, describe your process of learning how to use CoSpaces. What was familiar? What was challenging? How does it feel to be learning something new? How did you react or deal with challenges or roadblocks?

When I began learning CoSpaces, I started with watching the recommended videos. I've never used CoSpaces before so starting with the videos gave me a better understanding of the software. While I was watching the videos, I had CoSpaces open but I didn't interact with it because I wanted to see exactly how CoSpaces worked from someone else's perspective. The videos were really helpful because I was able to learn from the instructor without jumping right in and pressing random buttons.

After watching the videos, I replayed each video but while I was watching, I interacted with the Free Play version of CoSpaces. I realized that with each interactive software, there's shortcut buttons for the essential segments such as 'a' to attach the object to another object. The shortcut keys in CoSpaces are very reliable as I'm not only utilizing the pointing device to do everything; it makes working with CoSpaces a lot easier. Also, I found it familiar that each software has built-in products that I can use such as the environments and library while I can also place in object files of my own.

While I was using CoSpaces, I found it challenging when I would add more objects to the drifting course. CoSpaces would start to glitch and be more difficult to work with when there's more components placed in the world. At times, I would need to reload the page, take animations out of the world and minimize how much I can use to make the program run more smoothly. Though it was challenging to move around the drifting course because of the efficiency of the program, it felt really good to learn something new! As I've been more fond of programming from other classes, I thought the coding was easier to work with and the same concepts as coding in other languages because it's still algorithms that are being executed.

To handle the challenges, unfortunately I had to limit how many objects and animations I was utilizing to make the program run better. At times, I would get on my desktop to finish the drifting course instead of using my laptop. When I used my desktop, it would run a lot more efficiently. Also, I would make other objects smaller to maximize the space. Overall, I would get past the roadblocks from minimizing the objects the best I can while maximizing the space CoSpaces was allowing me to use.

3. In 3-5 paragraphs,describe your programming experience and how it relates to using drag and drop code.

The programming experience that I have are from classes such as Starting Computing, Data Structures, Computer Systems, Computer Animation, Fundamentals of Human Computer Interaction, Algorithms, Principles of Programming Languages, and Software Development Methods and Tools. Throughout each class, I noticed that every programming language is heavy on the concepts and the syntax. If I understand the concept of programming, I can utilize the syntax of the provided programming language and create the algorithms I desire. Understanding the concept of an algorithm is essential for knowing how to thoroughly work with each language without having to read the 100+ page documents each time I want to use the language.

In relation to the drag and drop code, I noticed that the drag and drop code will execute one step at a time. The drag and drop code is very limited by how much we can actually do in the automata world while in coding such as computer animation, the algorithms we're able to place seems far from limited. Utilizing the drag and drop code in our automata world allows us to create actions such as riding a bike or driving a car. The algorithms we're able to create are built-in coding which is also a limitation.

Coding with the drag and drop code also relates to programming in general because of the recursion that's placed into our code arsenal. The loops that are in CoSpaces are similar to writing a while/for loop in such programming languages as data structures and software development. The actions in the code are similar to creating an input algorithm for the user to interact with. In the built-in code, there's two options of input answers being the correct answer and the incorrect answer which is the same as writing an input string, variable, etc. for the user to input and the computer will respond just as in starting computing.

CoSpaces is much more simple than understanding algorithms in Computer Systems. CoSpaces is very high-level programming while Computer Systems is very low-level programming. Understanding low-level programming made high-level programming easier to work with as I don't need to know every instruction that's occurring in the software. In all, both levels of programming are related because it is essential to know each algorithm that is being utilized when the hardware and software begin to execute the program.

4. In 3-5 paragraphs, on this page, how is your Digital Automata like a mechanical automata? How is it different? How is it the same?

The relations between a digital automaton and a mechanical automaton have similar and different components that need to be involved for us to create the desired automaton. A mechanical automaton has gears and components inside of the machine that all need to be working in order for the mechanical automaton to work efficiently. A digital automaton has animations and coding inside of the preferred object that all need to be working in order for the digital automaton to work efficiently. Both automatons are vital for having every favored component working correctly for the automaton to work properly.

A mechanical automaton has components such as a hand-powered element, motors, gears, wires, materials, transistors, etc. to make the automaton work as the creator wants it to. A mechanical automaton needs many physical components in order to make the automaton. For the mechanical automaton to work correctly, the creator would need to have the components inside of the automaton to be of the right size, length and quality necessary to make the desired mechanical automaton.

A digital automaton has components such as coding, animations, materials, speech, transformations, etc. to create the desired automaton. The coding that is placed inside of the automaton will make the automaton work as it's meant to work, given that the code is correctly written. A digital automaton can be resized at any length and width in just seconds. Also, it can be a different color and have different animations in a very little amount of time. Being able to change components at an ease makes the digital automaton effortless to work with.

A mechanical automaton is different from a digital automaton because a digital automaton only needs the program in order to work while the mechanical automaton needs physical components in order to work. A digital automaton can change so many aspects in the matter of seconds while it would take much more time to change the aspects of a mechanical automaton such as it's size, color, and animation. Furthermore, specifically changing the animation of a mechanical automaton would be highly time consuming while a digital automaton would take a little amount of time to revise. Though, both automatons have set algorithms that are executed when the programs are running. From the time the mechanical automaton is turned on and the digital automaton has started to the end of each automaton, the algorithms are running as desired.

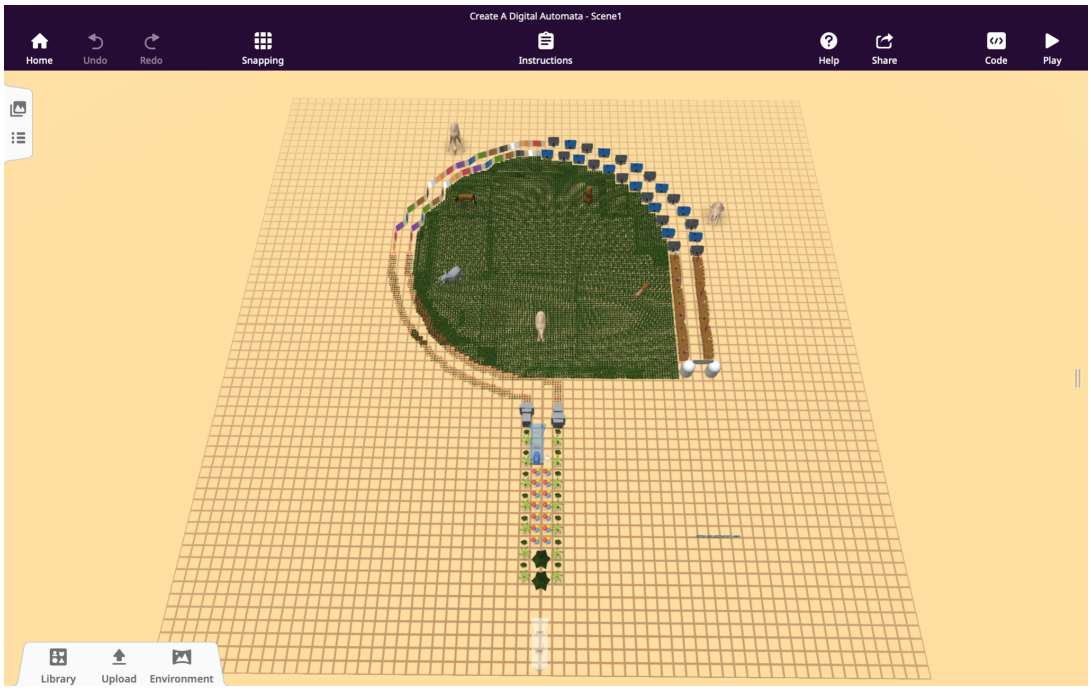
5. In 3-5 paragraphs on this page , consider the metaphor " Mind is to brain as software is to hardware in a computer." Discuss this statement in relation to your Digital Automata.

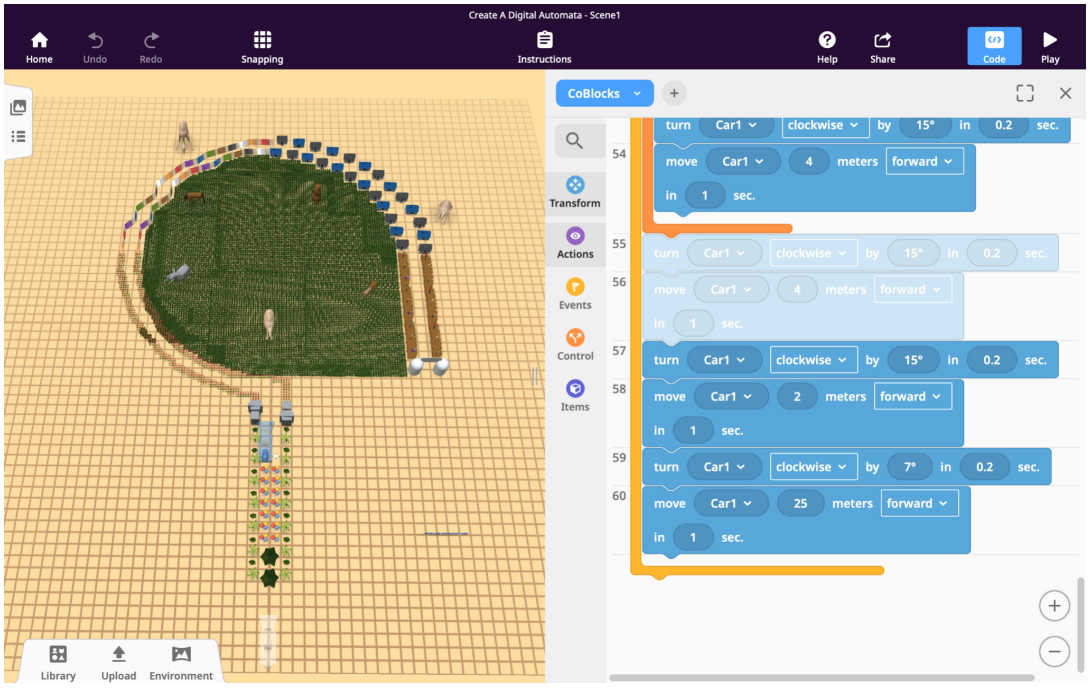
Mind is to brain as software is to hardware in a computer are related because as our minds learn about objects around the world and/or situations that we come upon, the mind is the software that's doing the learning while the brain is the hardware that's containing the knowledge. Our minds work similar to a computer because we process information as a software would and need our brain to run the information as hardware would. Our brain receives information from the world around us and we mentally create algorithms to understand the world from our brain.

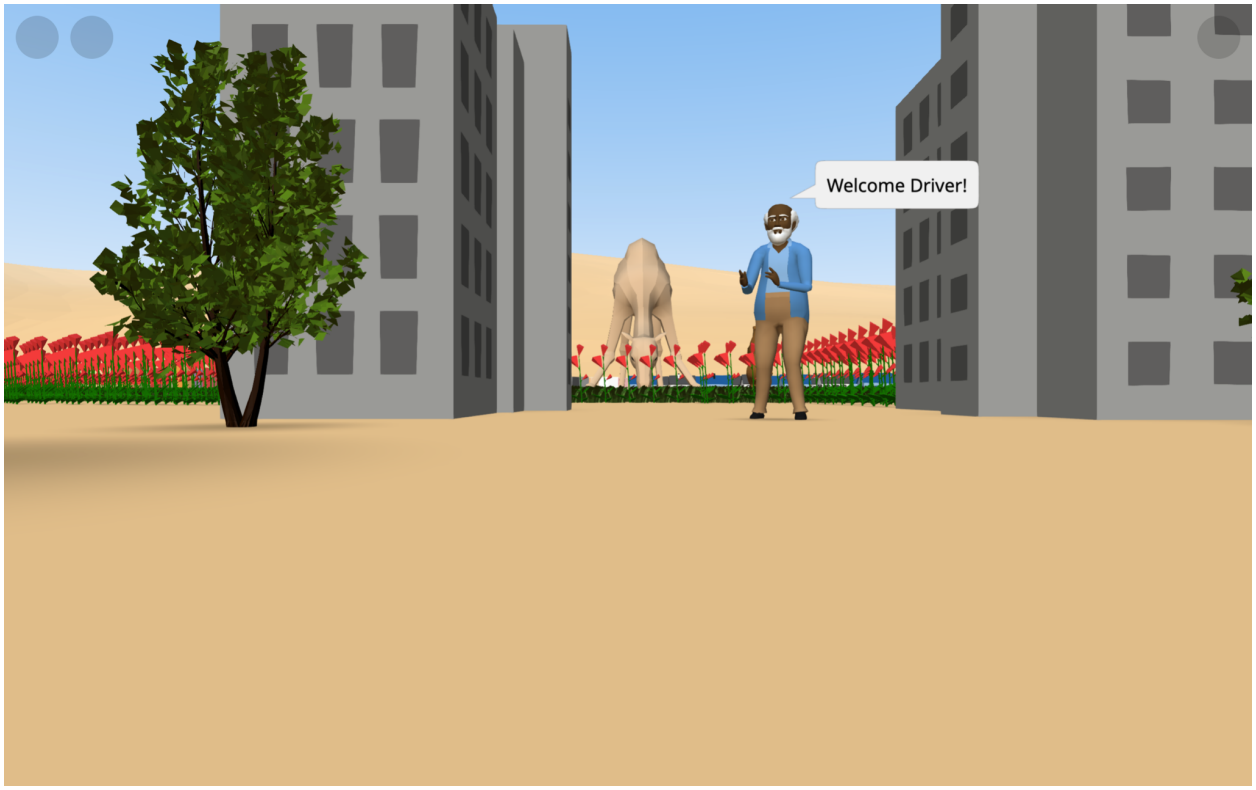
In relation to our Digital Automata, the algorithms that are in the software are processing and being held by the hardware of our computer. Our Digital Automata takes in objects, animations, materials, etc. and it's all within the hardware. The software is attempting to understand the algorithms that we create while the hardware is running the algorithms to the best of its ability. Our Digital Automata is in relation to the metaphor "Mind is to brain as software is to hardware in a computer" because the mind and software are receiving the information provided while the brain and hardware are running the information. Even with the front-end processing that our Digital Automata is doing, there's also background algorithms that are being utilized just as similarly as our brain takes mental notes of thoughts that aren't prioritized at the time, but are still important.

The mind is to brain as software is to hardware in a computer are connected because all functions of the mind work together in the brain such as the software works together with the hardware. The mind and software are connected because they are both incorporeal. The brain and hardware are connected because they are both physical elements. Mind is essential for the brain just as software is essential for hardware in order for each component to successfully work. Our Digital Automata is a representation of software and hardware working together as we place algorithms into our world and have the hardware run the algorithm efficiently.

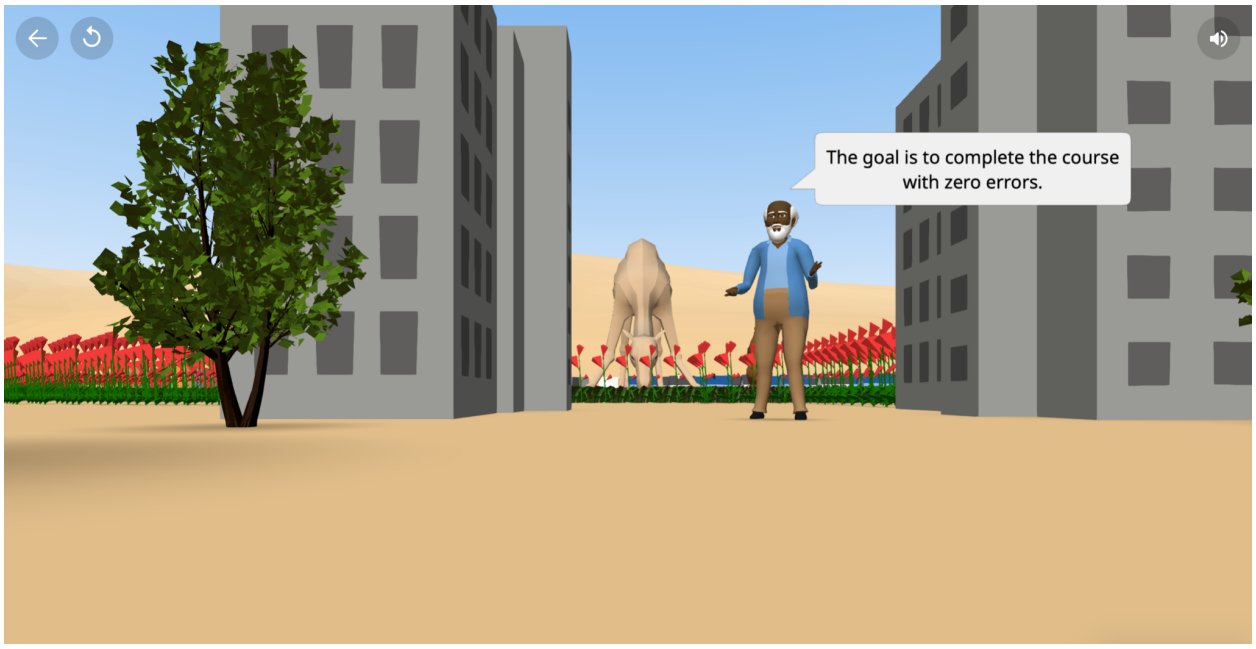
6. Adding as many pages as you like, include some screenshots of your Digital Automata.

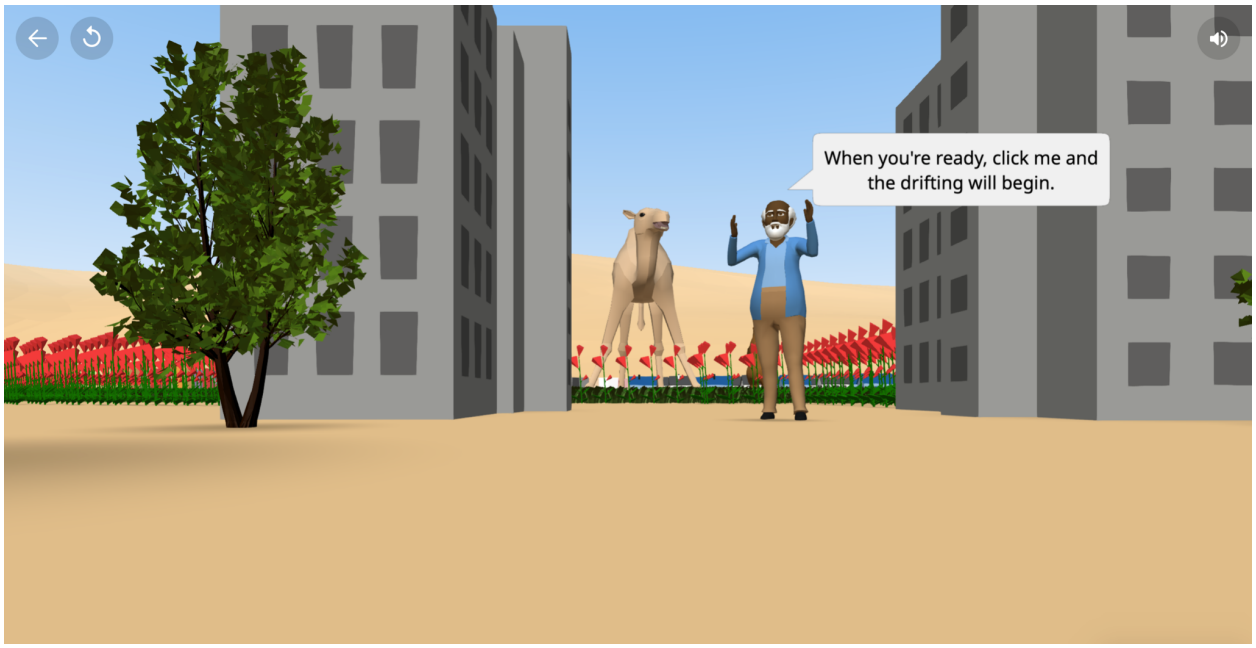


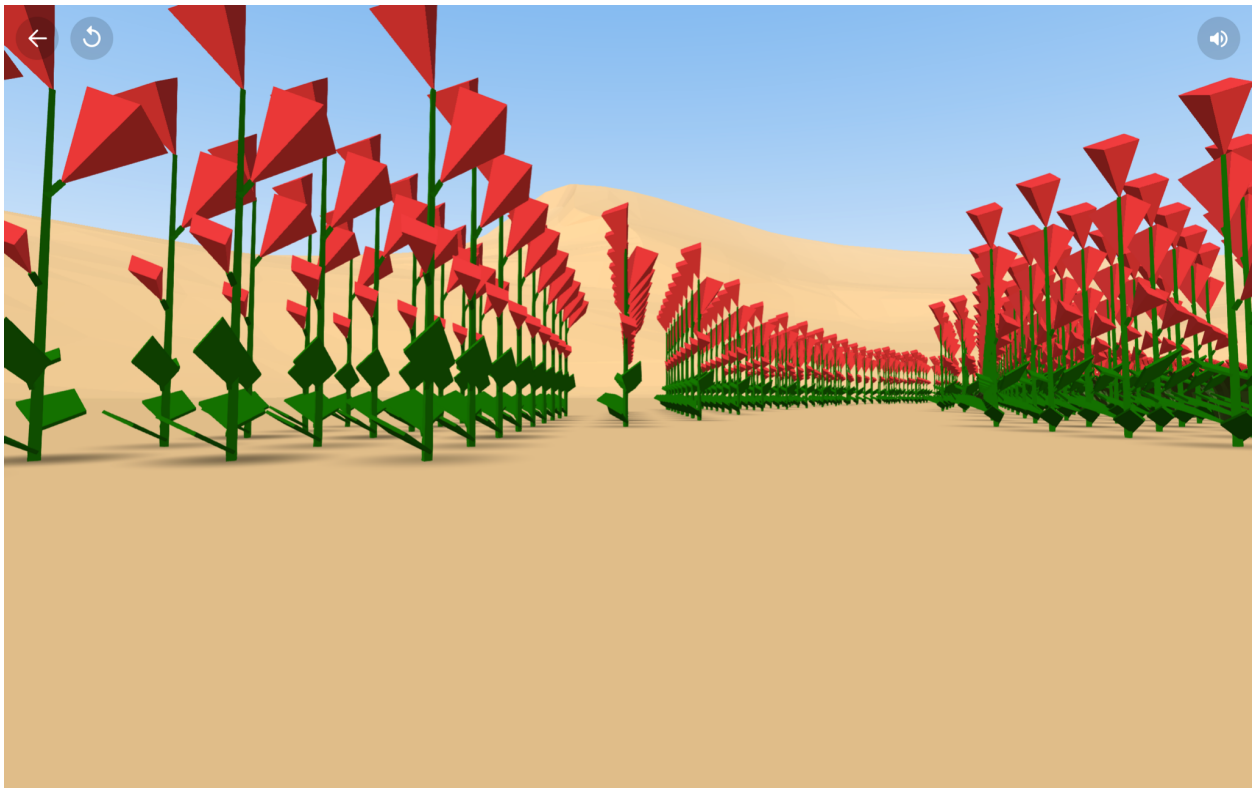


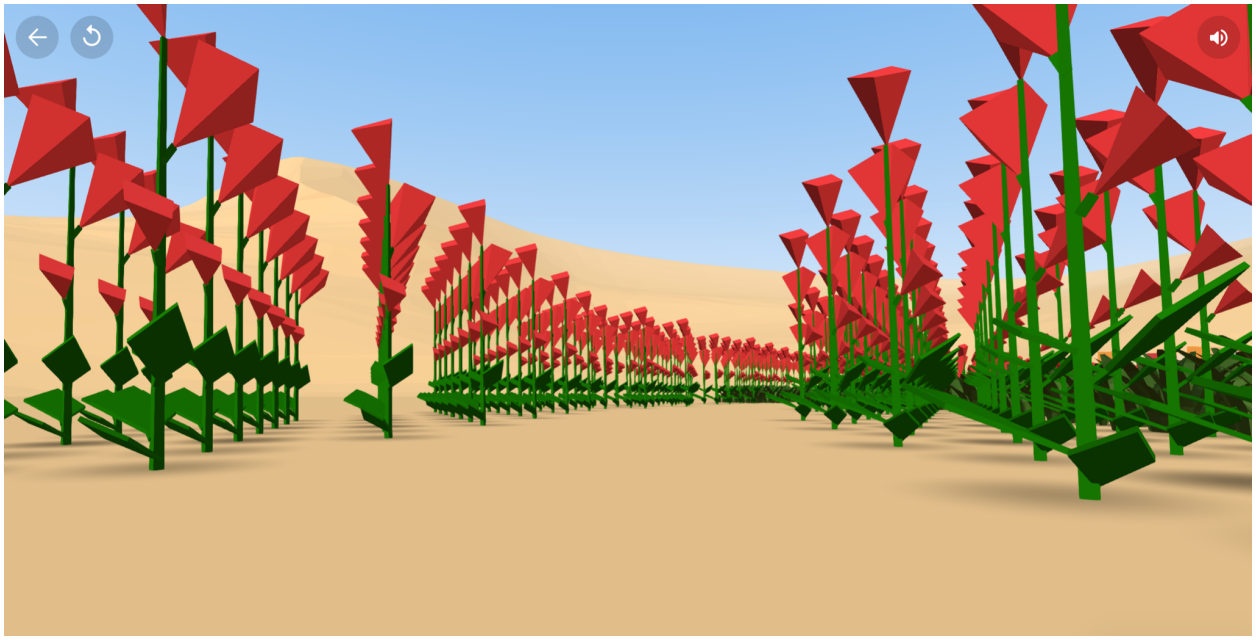

Welcome Driver!

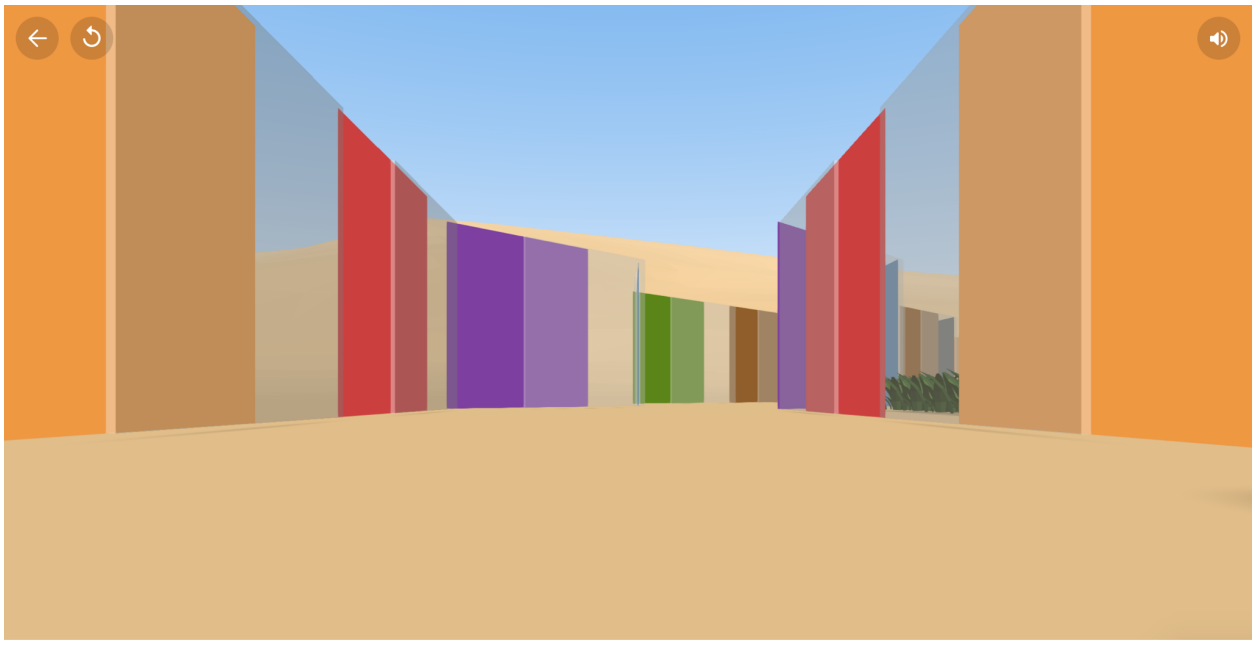

My name is Papa Drift King and this is my drifting course.

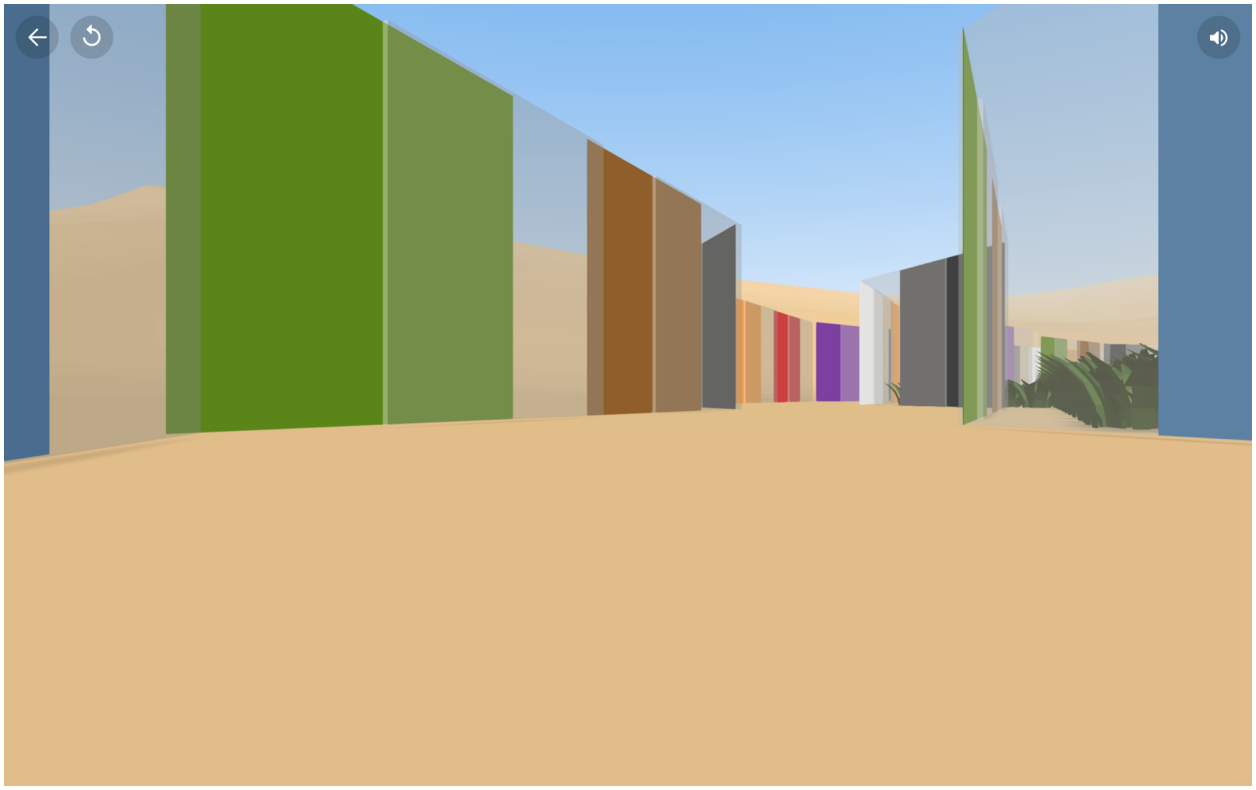

The goal is to complete the course with zero errors.

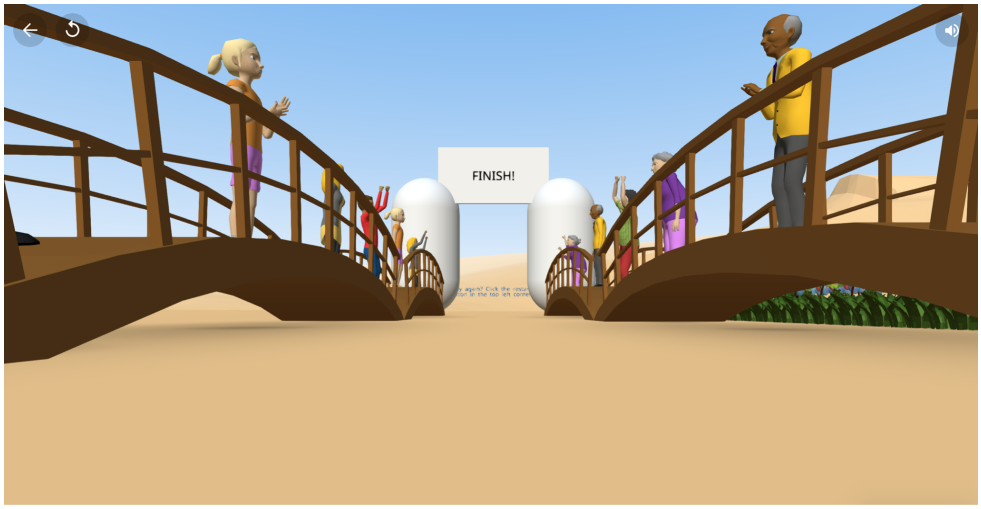

When you're ready, click me and the drifting will begin.










FINISH!


Play again? Click the restart
button in the top left corner.