A register machine is a theoretical computer that consists only of memory locations, and a processing unit. An instruction register stores the instruction to be executed, and these instructions represent computational functions like add and subtract. Each register machine has a specific set of instructions it can accept.

This project is updating a register machine simulator, called Rodrego, so it can run on modern computers. RodRego has only 2 instructions – increment, and decrement/branch, which work on a specific, named register within the instruction.

Step	Instr	Reg	Go to	If 0, branch to:
1.	Inc	1	2	
2.	Deb	2	1	3
3.	End			

This program increments register 1, then decrements register 2. When register 2 is empty, the program branches to Step 3, which happens to be "End". This program, therefore, adds register 2 to register 1.

This project requires a visual interface which is clear and conveys the concept of a register machine to someone who has no knowledge of how a CPU works or what a register machine is.

1) Line Application
   The coloured boxes represent registers, and the line represents the program.
   The coloured boxes can be edited to enter data into the register.
   The empty box can be dragged to the line – either above, or below. It will be tapped to change colour to match the register. If the box is placed above the line, the register will be incremented at that step. If the box is placed below the line, the register will be decremented. Therefore, each box represents exactly one instruction that will have action on one register.
   The user will only be able to place the box above or below the line, next to the previous instruction.
   There are two kinds of arrows – the "backwards" arrows, above the line, represent a repetition. Once the instruction is completed, the program will follow the arrow back to the instruction that is being pointed at, and will repeat this, and any others within the arrow span
   The forward arrows, below the line, are the branches for the deb instruction. This is where the program should go when the register is zero.
   Finally, there is an "end" symbol. This is placed below the last instruction. Multiple of these can be placed – for example, if the program should end if a register does not have a correct start value the user may want the program to cancel.