

## Reaction

### Challenge: Keep score

The key to this challenge is using variables. You can think of a variable as a bit of code that remembers a number which might change during your program. They're great for using when you need to keep count of something like a score.

From "Variables", click "Make a Variable" and it will ask you to give a name for your variable.

You'll need two variables - one for each player. You could call them **playerA** and **PlayerB**



It's a good idea to set up variables before anything else happens in your program, and we can do this by dragging a "Set Item to" block into the "on Start" block.

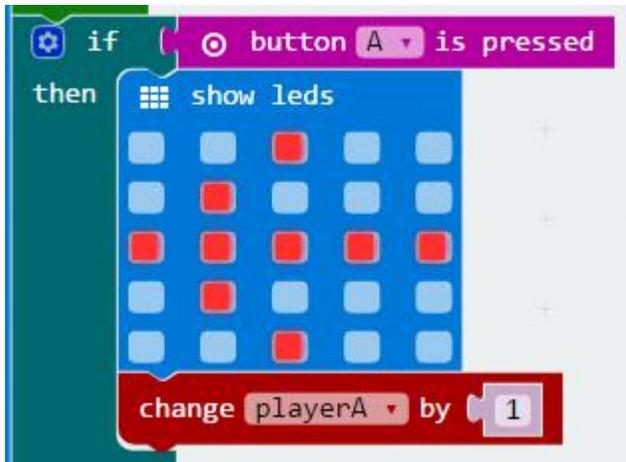
Because you have two variables - one for each player's score - you'll need two of these blocks.

You'll also need to change the block show the name of one of the variables we just set up.



Now you need to increase the score by 1 for the player who pushes their button fastest.

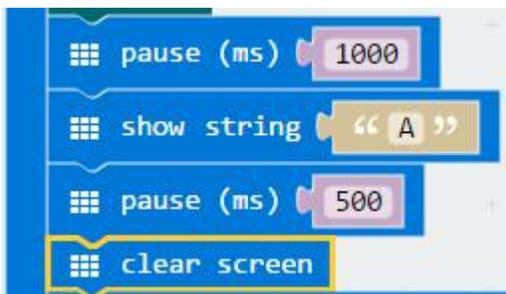
Add a "Change Item by 1" block to the end of the "If button A is pressed" loop, and change it so it uses the playerA variable.



Repeat this step so the “If button B is pressed” loop changes the playerB variable.

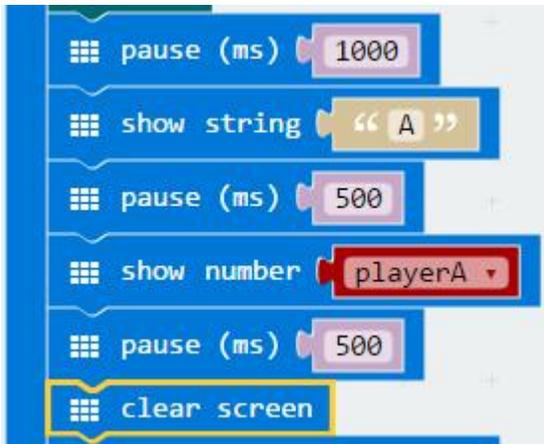
OK, so your program now adds 1 point each round to the score of the winning player. But it doesn't tell you what the scores are yet, so you need to add a bit more to get it to do this. We'll do this at the end of the code.

Between the “Pause” and “Clear Screen” blocks at the end, drag a “Show String” block and a “Pause” block. Change the string to say “A”, and the Pause to 500



Next drag in a “Show Number” block and another “Pause” block.

You need to change “Show Number” so it shows the score for Player A, so from “Variables” drag in the “playerA” block.



Repeat those last 2 steps so you also have blocks showing the letter “B” and the score for “playerB”.

Now download your program to your Micro:bit and at the end of each round it will tell you what score each player has!

# Reaction

## Debugging

Some times your code might not work exactly how you thought it should and might do something unexpected. These problems with your code are often called “Bugs”, and fixing them is sometimes called “Debugging”.

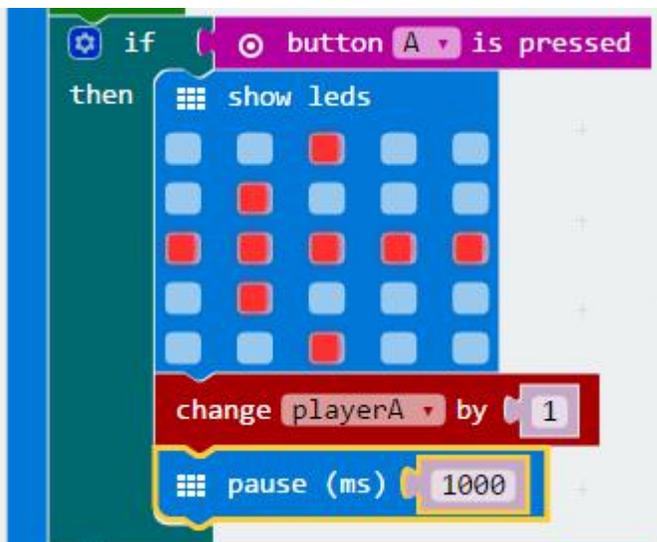
For example. You might notice that sometimes it thinks more than one player wins a round - usually it will think that after Player A has won, Player B also then wins the same round. But you can fix this!

Here are two ways you might fix that bug - a quick way and a reliable way. You’ll be doing the quick way here, but you might want to try the reliable way later on - you can find it at the end of this worksheet.

### 1 - The Quick Way

First, take out the “Pause” block from just above the “Show string “A” block you just put in.

Next, add a “Pause” block on to the “If button A is pressed” loop and set it to 1000



Now repeat this for the “If button B is pressed” loop.

Download the program to your Micro:bit and it should stop counting 2 winners for one round.

Why did you do these last steps? Well, the way the program is made, it checks to see if button A is pressed, then checks to see if Button B is pressed just after (it does this very very quickly though - faster than you would be able to notice). This means that sometimes if player A wins the round but lets go of their button before player B does, it thinks player B has *also* won during that round. What you did was add a pause of one second each time it detects a button press and adds



Willitwork.co.uk

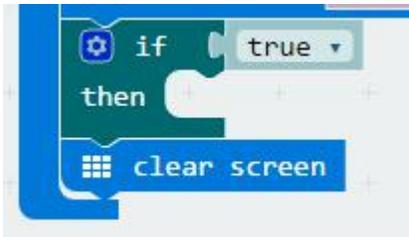
to a player's score - this should hopefully be enough time for both players to have let go of their buttons. If not you can always make those pauses longer!

## Reaction

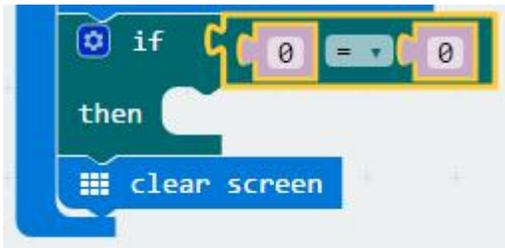
### Additional Challenge: The Winner is...

During the Library Micro:Bit sessions, one of our coders - Oliver - had a great idea for another challenge for the Reaction game. He wanted the first player to score 10 points to be declared the winner. So let's have a look at one way you can do this!

From "Logic" drag an "If... Then" block and snap it in just before the last "Clear Screen" block in your code.



You want the program to check how many points PlayerA has scored, so from "Logic" drag an "equal" block and snap into the "if" section.



Now you need to change it to tell your code to check if PlayerA has scored 10 points yet.

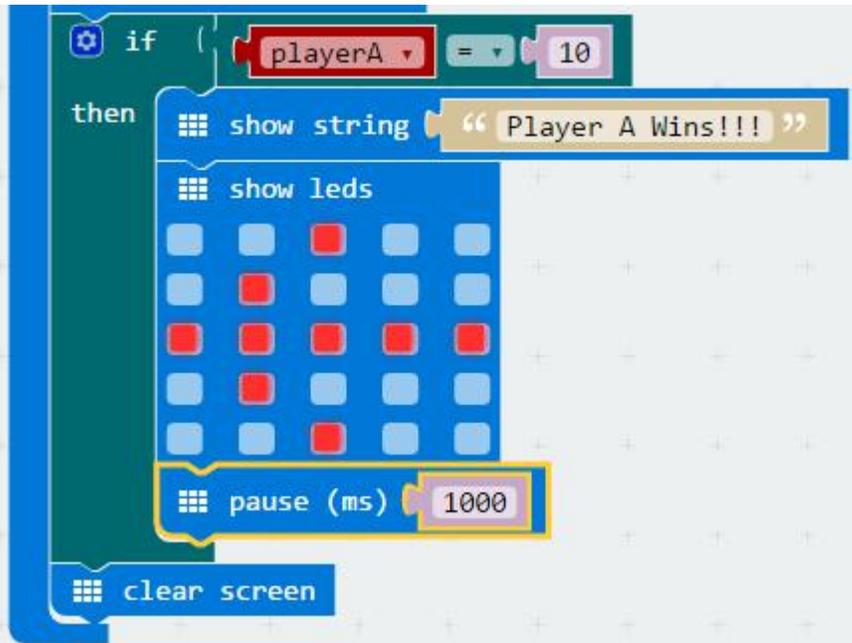
From "Variables" drag a "PlayerA" block and put it in the first part of the equals block. Change the second part of the equals block to "10"



OK, so your code knows you want it to do something when PlayerA's score equals 10. But you need to tell it what it should do. We can do this by putting more code into the "then" part of the "If... Then" block

Drag in a “Show string” block and change it to say “Player A Wins!!!”

Also drag in a “Show LEDs” block and use it to draw an arrow pointing to playerA, then put in a “Pause” block and set it to 1000.



So what you’ve told your Micro:bit to do is

- at the end of every round, check if Player A has scored points
- If they have, show a message on the LED screen saying Player A wins, then draw an arrow pointing to Player A and wait for 1 second before carrying on.

If playerA has **not** scored 10 points the code will just skip the “then” section of the “If... Then” block.

You need to make the same sort of thing for Player B, so drag in another “If... Then” block beneath the one you’ve just made and run through the same steps for player B.

The image shows a Scratch script with the following blocks:

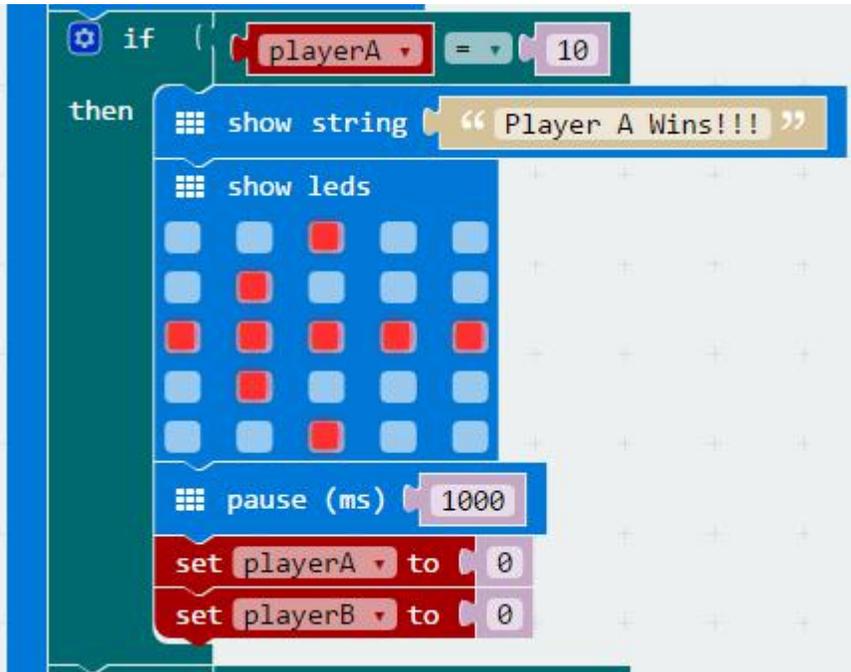
- if** (playerA = 10) **then**
  - show string** "Player A Wins!!!"
  - show leds** (5x5 grid with red and blue squares)
  - pause (ms)** 1000
- if** (playerB = 10) **then**
  - show string** "Player B Wins!!!"
  - show leds** (5x5 grid with red and blue squares)
  - pause (ms)** 1000
- clear screen**

You can now download your code to your Micro:Bit and when you play the game it should tell you who wins by getting 10 points first!

But wait - you might notice that after someone wins the game carries on and the players keep the points they'd already scored. In fact, when the other player reaches 10 points the game will declare them the winner too!

Maybe we need to reset both the player's scores back to 0 at the end of the game?

You can do this by dragging 2 “Set item to” blocks into the end each “If... Then” block, and set them so one makes the PlayerA variable 0 and the other one makes the PlayerB variable 0.



Drag 2 more “Set item to” blocks to do exactly the same with the “If... then” block that checks player B’s score.

What you’ve done here is tell your Micro:bit that every time someone gets to 10 points and wins, after it’s shown you the Winner’s message it also needs to reset all the points to zero. If you remember when you first made some variables to keep track of the player scores, you put 2 blocks into the “On Start” section of your code - they made sure both players had zero points at the start of the game and this is really just doing the same to reset everything to the beginning.

If you’ve got this far with your code then Well Done! You’ve learned some **very** important things here about using variables to keep track of numbers (like scores, or how many times something happens), and using Logic to get your code to check if something special has happened and then do something else *only* if it has happened. That’s a really big deal in programming because once you can make a computer remember things and decide what to do when particular stuff has happened, you can make programs that react to all kinds of things happening!

# Reaction

## Debugging (again)

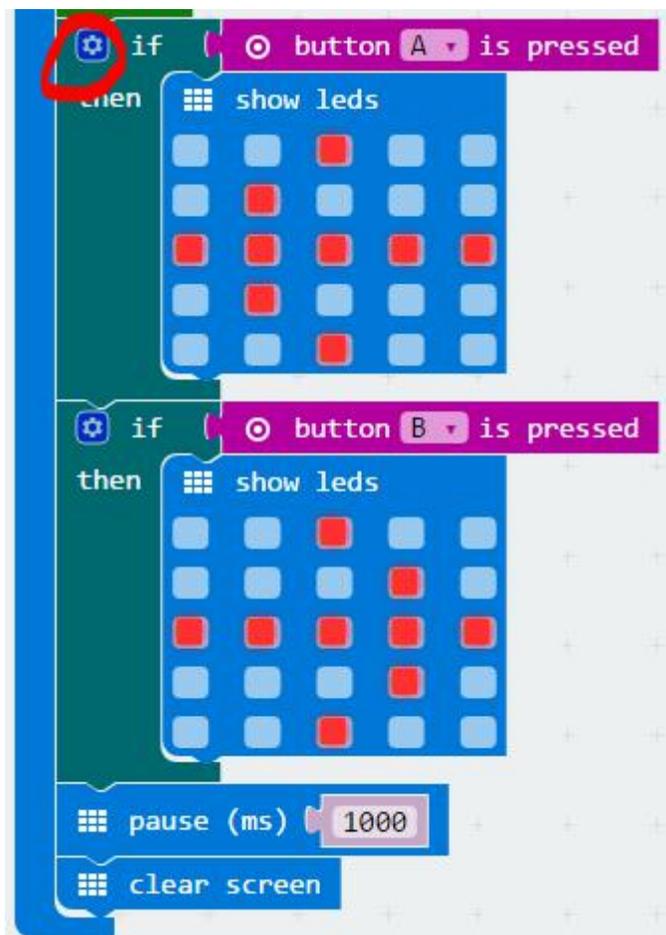
This is a second way to fix that bug where more than one player could win in a round. Give it a try if you want!

### 2 - The Better Way

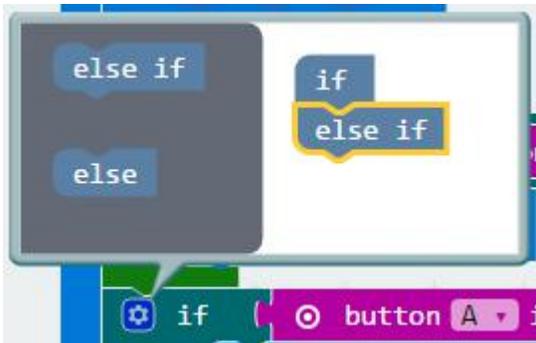
Really the problem with the code is that it's doing 2 separate checks - there's one block to get it to check for Player A pressing a button, and then a second block that runs straight after that checks for Player B pressing a button.

But what if we could make just one block that can do both checks for both players?

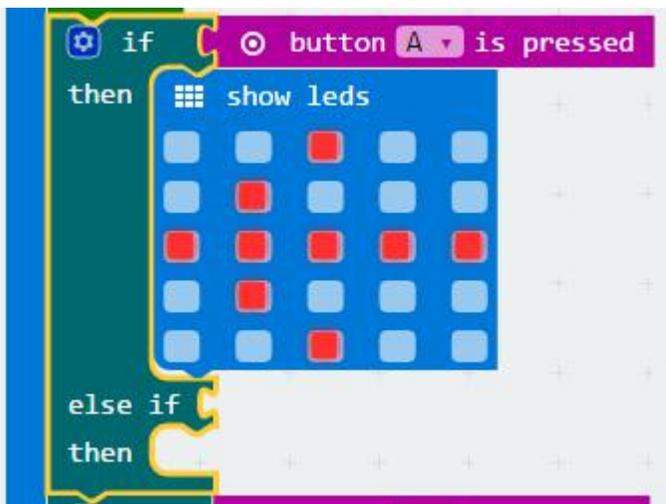
Well, you can. Let's pretend you've just finished the code on the Code:Club worksheet and have not started the "Keep Score" challenge yet. If you look at the first "If... Then" block, you can see a little cog in the top left corner. This lets you change the block to it can do extra checking of things.



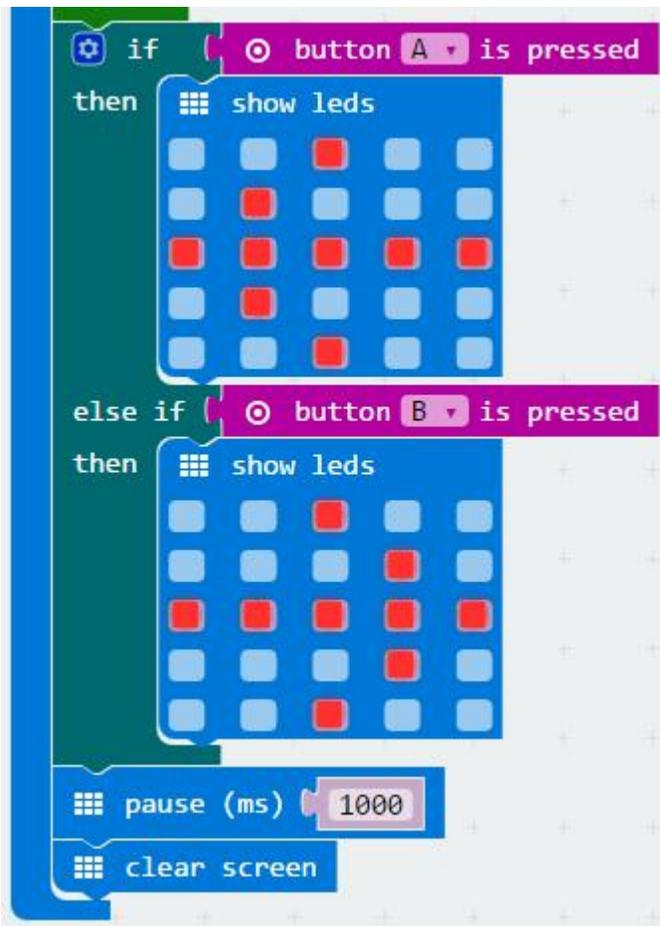
Click on it and a little box will open up. Drag an “else if” block from the left side and clip it on beneath the “If” on the right side.



Once you’ve done that, click on the cog to close the box again. If you look at your code now, you’ll see that the “If then” block for Player A now has space to check for an extra thing.



What you can do now is drag everything from the “If button B is pressed” section into the same one as the button A check. Once you have done that there will be an empty “If then” block where the button B stuff used to be, so you can delete that.



Now if you load your code onto your Micro:Bit, you should find that it doesn't let more than one player win in any of the rounds!