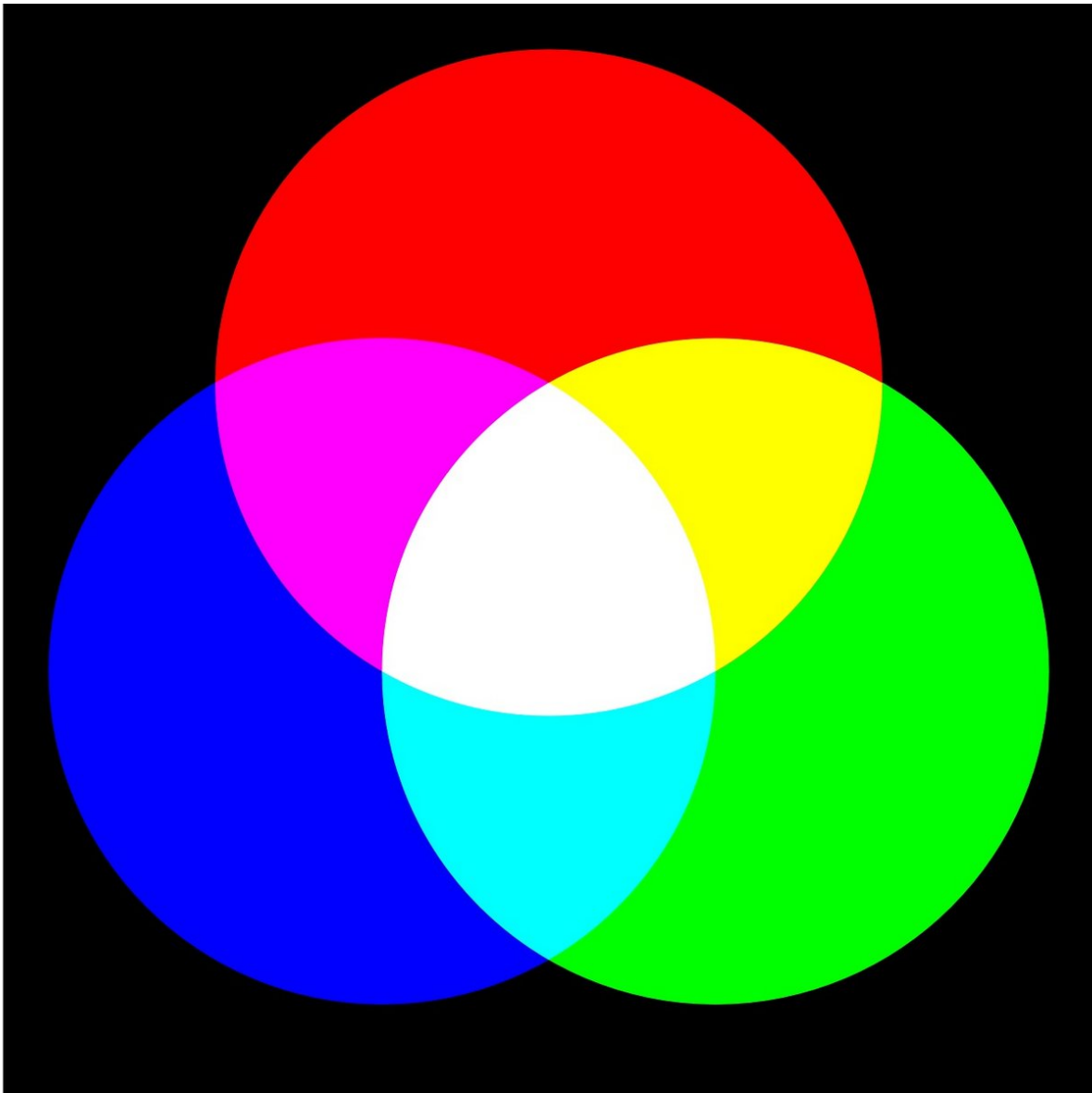


C - Colour Mixing

Let's use the Sparkle module and some switches to make a colour mixer!



INTRODUCTION

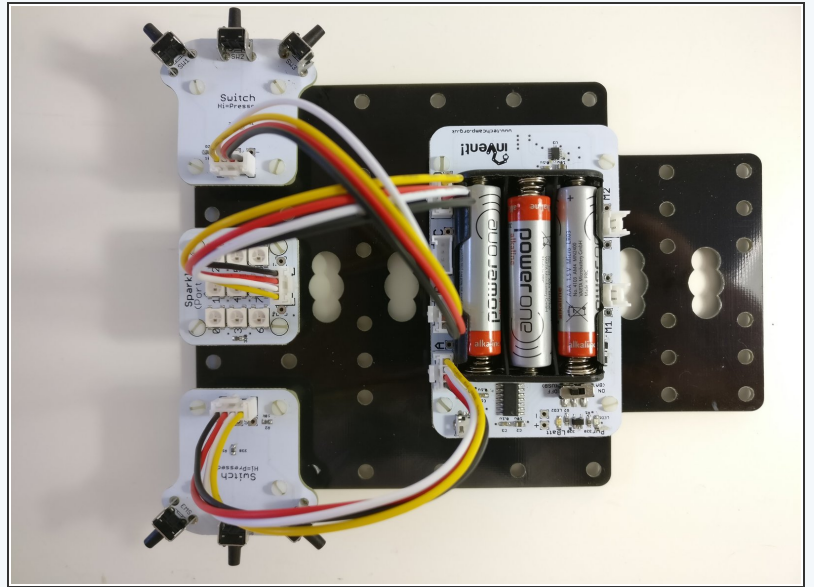
Let's use the Sparkle module and some switches to make a colour mixer!

Step 1

Colour Changing Light

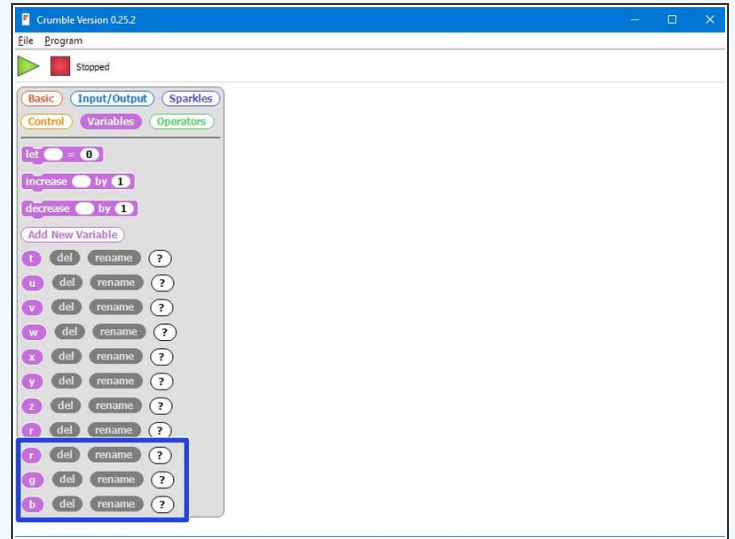
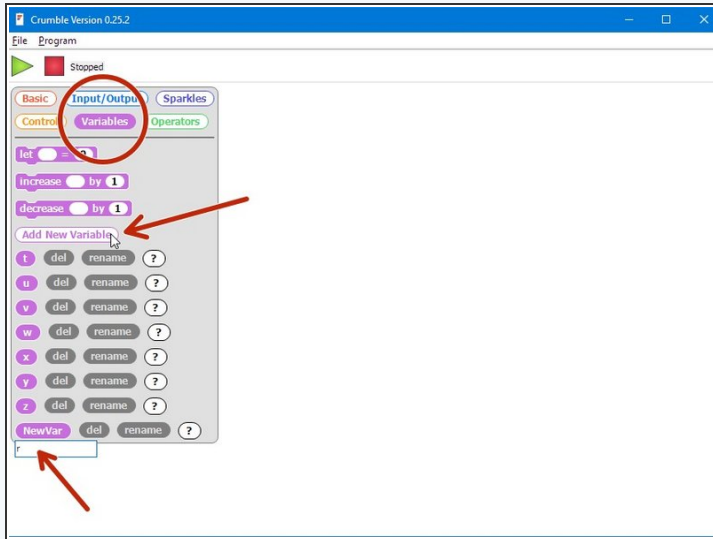
- Remember how each Sparkle contains **3** different colour **LEDs**?
- We're going to make a program using **variables**, that allows you to change the Sparkle colour using **2 switches**.
- **Assemble your robot** like the picture first!

⚠ Don't forget, Sparkles always need to be plugged into **D**.



Step 2

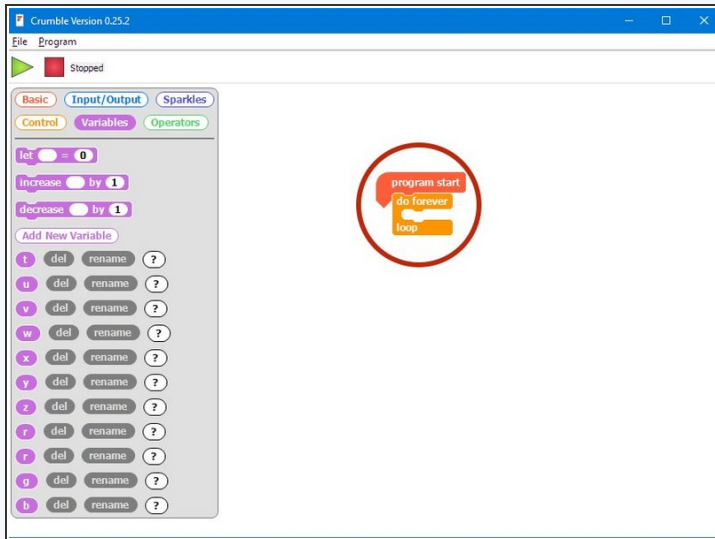
Setup the Variables



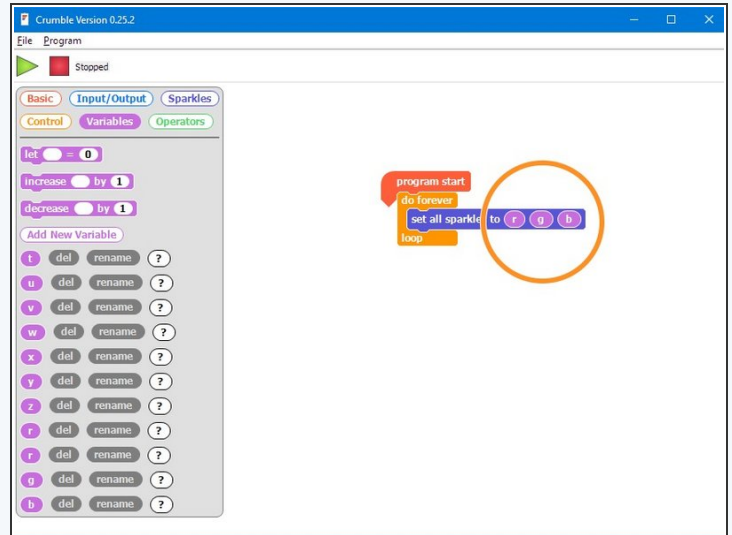
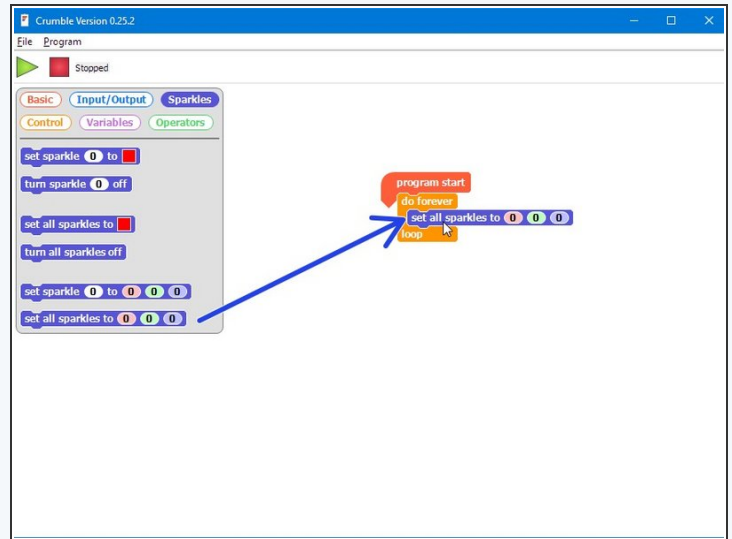
- Let's setup the start of our program.
- We are going to need **3 variables** this time - one for **red**, one for **green**, and one for **blue**
- **Add three new variables** using the "**Add New Variable**" button in the **variables** menu.
- Call them **r**, **g** and **b** (for red, green and blue).
- We're going to use each of these variables to **remember** the amount of red, green and blue light we want.

Step 3

Start the Program

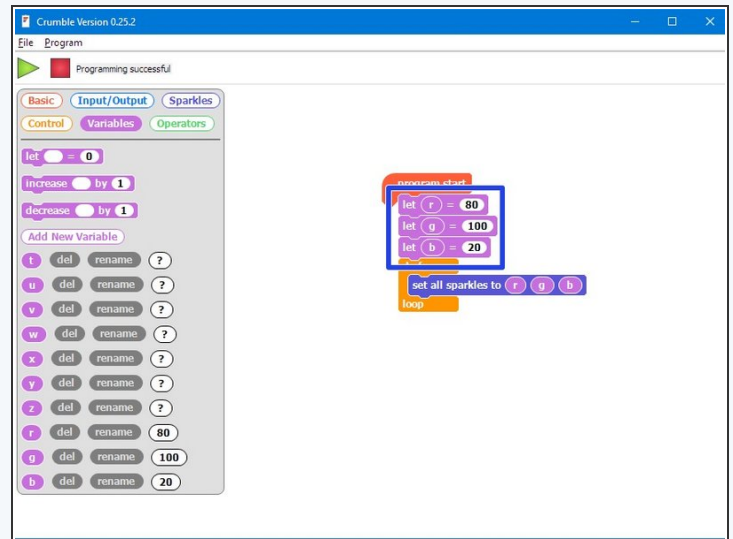
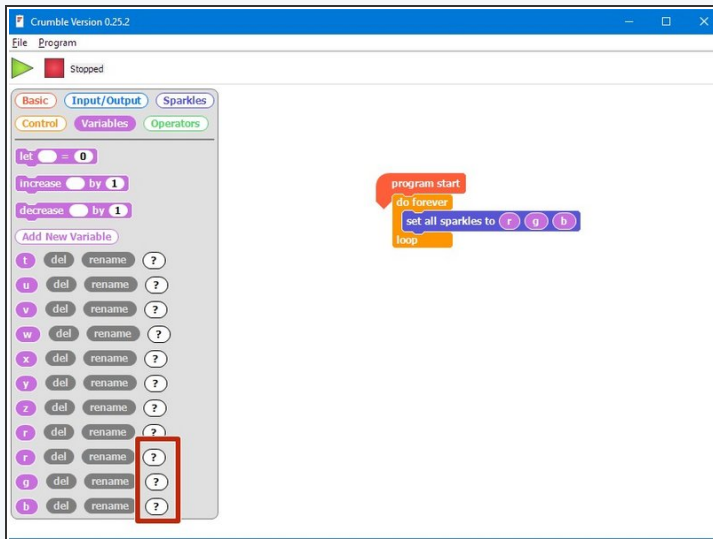


- Our program needs to always be **updating** the colour of the Sparkles, so when the variables change, the Sparkles change **too**!
- We will need a **do forever** loop for this - drag one into your program.
- There's another Sparkle block we haven't used yet, that allows us to set the amount of red, green and blue using **numbers** instead of the **colour picker**. **Drag one** into the do forever loop.
- Finally, **replace the three zeroes** in the sparkle block with the **r,g** and **b** blocks.



Step 4

Test it out!

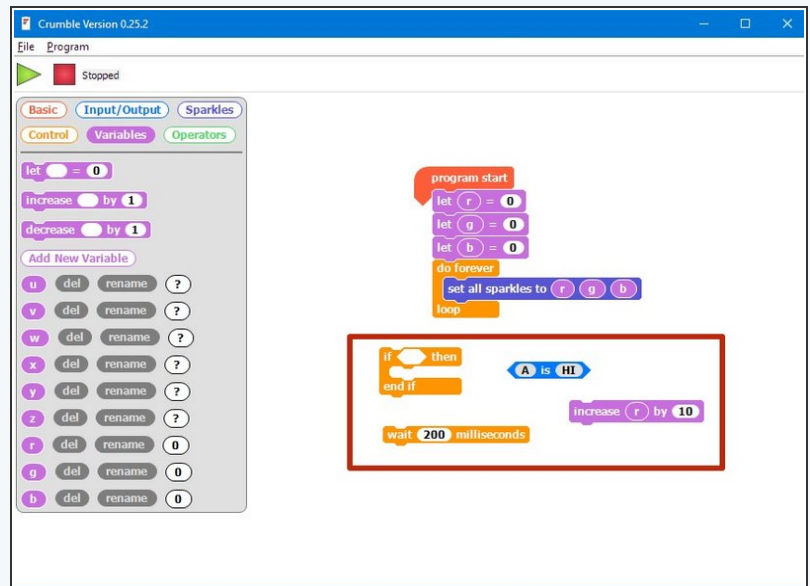


- We need to do **one more thing** before testing the program.
- If you look in the **variables menu**, r, g, and b currently have **no value at all** (that's why you see a "?").
- Before you use a variable in a program (like we have done in the Sparkle block), you need to **set it equal to something** - this is called **initialisation**.
- Pull in **three** blocks like the picture to **initialise** r,g, and b, **before** the **do forever** loop.
- Set them equal to some **different numbers** - **try it out** and see what colours you get!

Step 5

Using a Switch

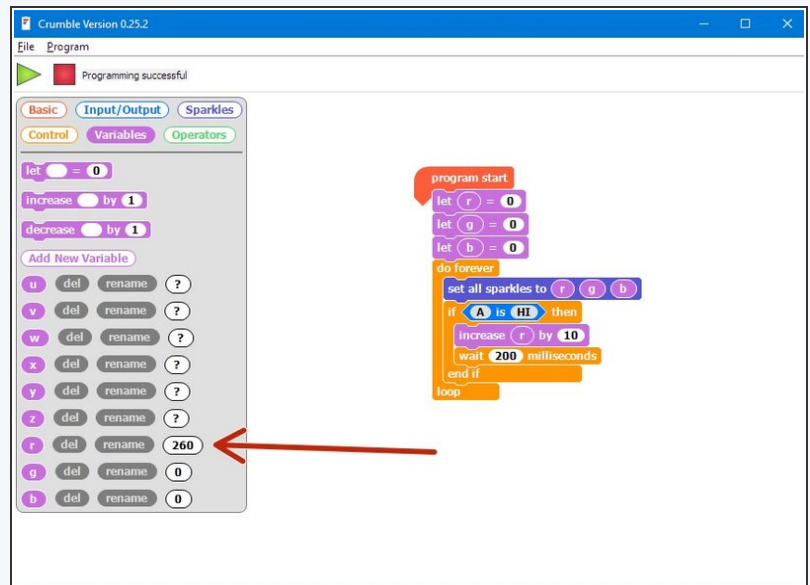
- Now let's add some blocks to let use **change the colour** of the Sparkles using a **switch**.
- **Inside** the **do forever** loop, add an **IF block** that checks if the **first switch** is pressed.
- If it is, **increase** the r variable by **10**.
- Add a **wait block** of **200** milliseconds after r is increased. Otherwise, holding down the switch will **change r very quickly!**
- At the start of the program, **set r, g and b to 0** again so you can see what is happening more clearly when testing your program.
- In the picture are the blocks you need if you want a **hint!**



Step 6

Back to 0?

- You probably noticed when testing your program that if you press the button enough times, the amount of red seems to **start from zero again**.
- This is because the amount of red, green and blue in a Sparkle **cannot be more than 255!**
- When r is **more than 255** (when you have pressed the button more than 25 times), the Sparkle will show the **value of r minus 255**.
- For example - 260 is **more** than 255, so the Sparkle will actually show **260-255=5!**
- **Watch** the value of r in the variables menu and **see what happens when it goes above 255**. Don't forget to keep your robot plugged in to do this!

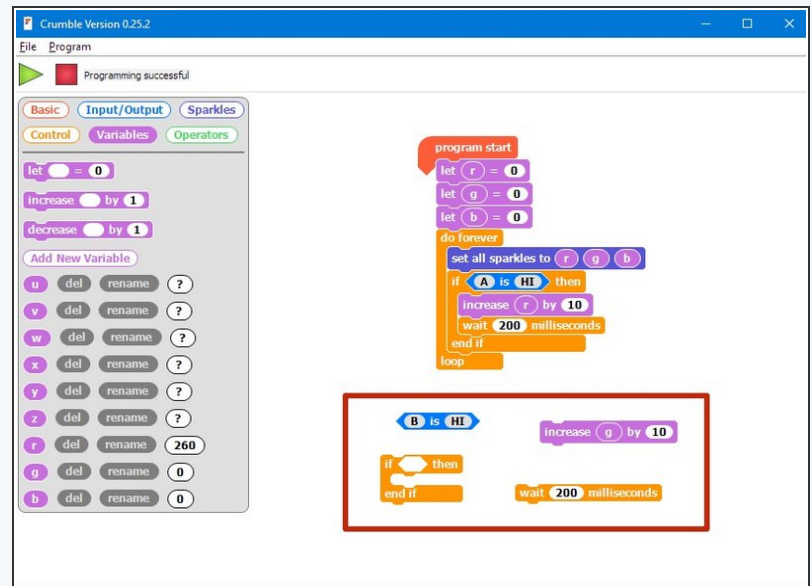


Step 7

Add Some Green

- We can now change the amount of red, but we want a colour **mixer**! Let's add **another colour** using the other **switch**.
- Add another **IF block** in the do forever loop, that checks the **second switch (B)** and increases **g** by **10** if it is pressed.
- Don't forget to use another **wait block**!
- There is another **hint** of the blocks required in the picture if you need some help.

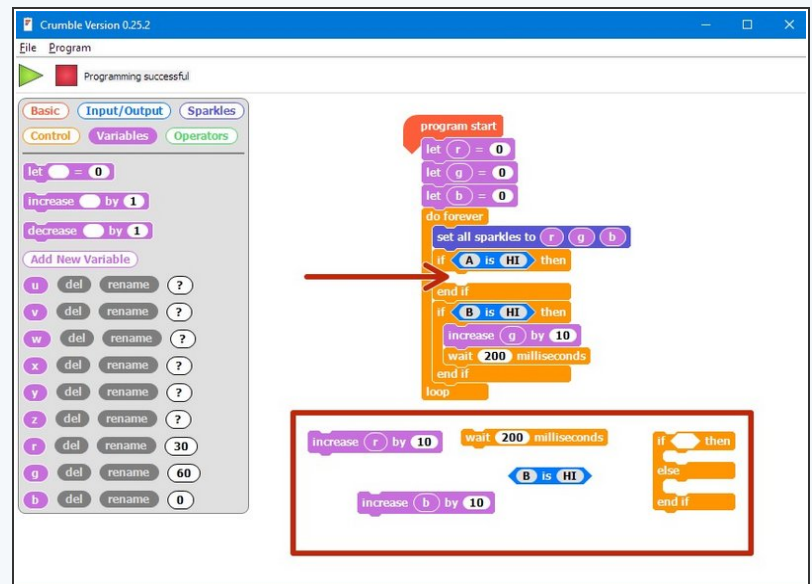
⚠ Make sure to **test** your program properly before moving on. What do you notice when you add equal amounts of red and green?



Step 8

3 colours, 2 Switches

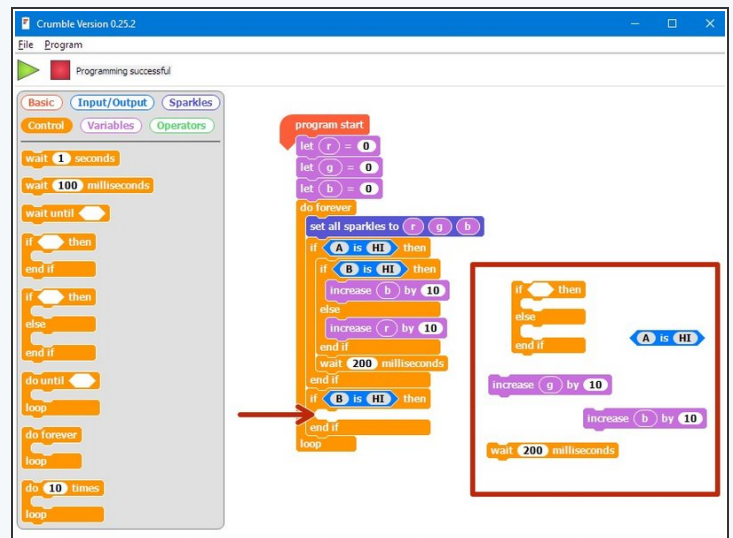
- You may have noticed that we have **run out** of switches for the **third colour**, blue - but there is a solution!
- What if we made a program that could increase b by 10 if **both switches are pressed at the same time**?
- To do this, inside the **IF block** that checks the first switch, put an **IF/ELSE** block that checks the **second** switch.
- **If** the second switch is pressed as well, **increase b** by 10 instead.
- **Else**, just increase **r** by 10 as before.
- Don't forget to **wait 200 milliseconds**!
- There are some more **hint blocks** in the picture if you need them!



Step 9

Fixing the two-switch blue/green problem

Challenge!



- You've probably found the last program doesn't quite work correctly - if you press both switches, the **blue and green both increase!**
- This is because the **second IF block** is still **true** if we are holding down **both switches**, so g is also increased by 10.
- Add an **IF/ELSE block** inside the **second IF block just like the one inside the first IF block** to fix this.
- Have a look at the **hint** if you need to!
- **Congratulations** - you have made a switch controlled colour mixer!

Step 10

White Light Buzzer

- When r, g and b are all the **same**, the Sparkles will be **white**.
- **Add blocks** to your program to:
 - Check if r, g and b are all the same (hint: you will need **3 IF blocks inside each other**)
 - If they are, **sound the buzzer!**
 - You will need to **add the buzzer** module and plug it into **C**.

