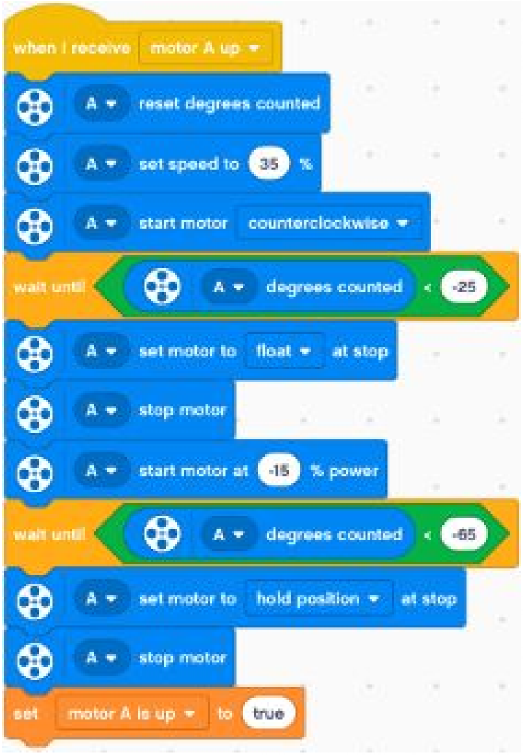
Conditions and Line Following

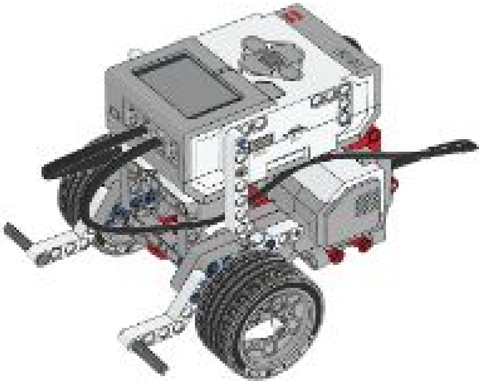
- Using “if” and “else”
- Color Sensor
- Color Decoder
- Following lines



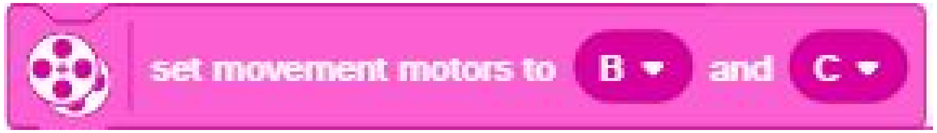
```
when I receive [motor A up ▾]
A ▾ reset degrees counted
A ▾ set speed to (35) %
A ▾ start motor [counterclockwise ▾]
wait until < A ▾ degrees counted < (-25) >
A ▾ set motor to [float ▾] at stop
A ▾ stop motor
A ▾ start motor at (-15) % power
wait until < A ▾ degrees counted < (-65) >
A ▾ set motor to [hold position ▾] at stop
A ▾ stop motor
set [motor A is up ▾] to (true)
```

A POSTERIORI
Play · Experience · Learn

EV3 vs Spike Prime

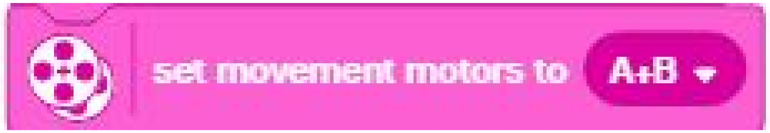


- We’ll be using the EV3 Classroom software for demonstration
- Programming blocks in Spike Prime are not exactly the same, but are very similar
- I’m sure you’ll figure it out! Let us know if you need help.



set movement motors to B ▾ and C ▾

EV3 Classroom

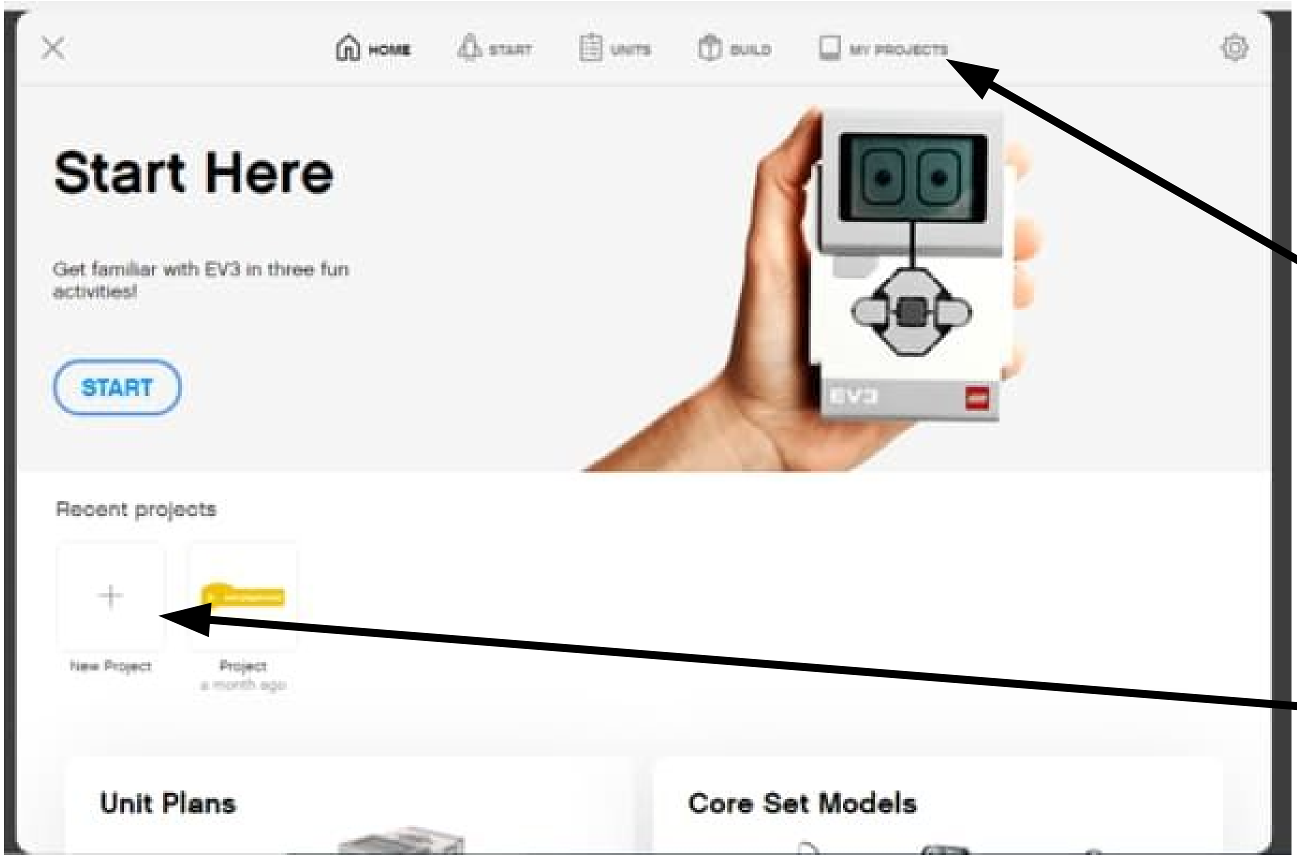


set movement motors to A+B ▾

Spike Prime

A POSTERIORI
Play · Experience · Learn

Starting a New Project



HOME START UNITS BUILD MY PROJECTS

Start Here

Get familiar with EV3 in three fun activities!

START

Recent projects

New Project

Project
a month ago

Unit Plans

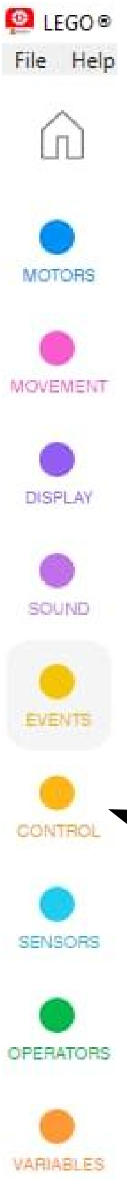
Core Set Models

Select “My Projects” to see your past projects

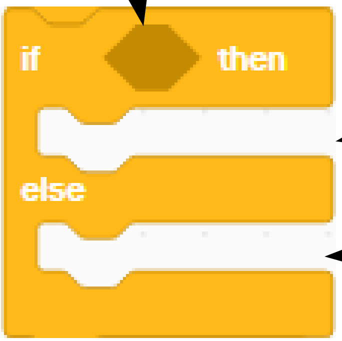
Start a “New Project”

A POSTERIORI
Play · Experience · Learn

“if” and “else”



This space is for the condition



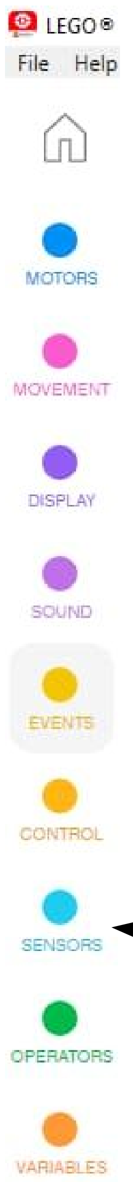
“if” the “condition” is true, then do this...

“else” do this...

Find it under “Control”

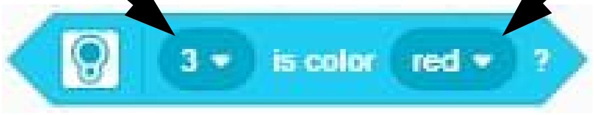
A POSTERIORI
Play · Experience · Learn

Color Sensor



Set the port number correctly (...check where you plugged in your color sensor)

Choose which color you want to test for



Notice the hexagonal shape of this block? ⬢

This is a condition block. It is either **True** or **False**.

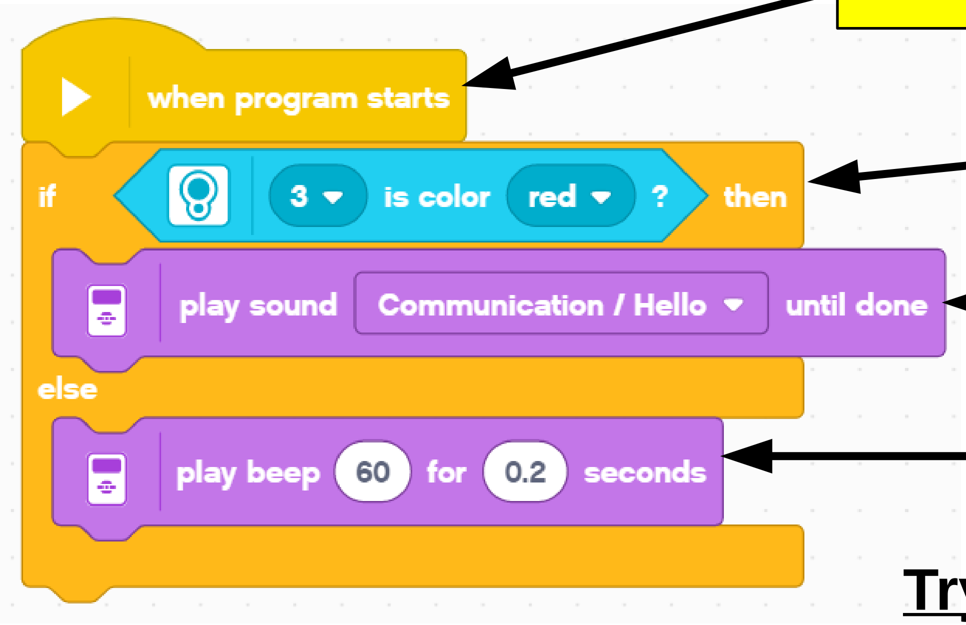
It fits inside the condition space of an “if / else” block.

Find it under “Sensors”

A POSTERIORI

Play · Experience · Learn

Example 1



```
when program starts
if [3] is color [red] ? then
    play sound [Communication / Hello] until done
else
    play beep 60 for 0.2 seconds
```

When the program starts...

Check if the color sensor is seeing red...

If it is, play “hello”...

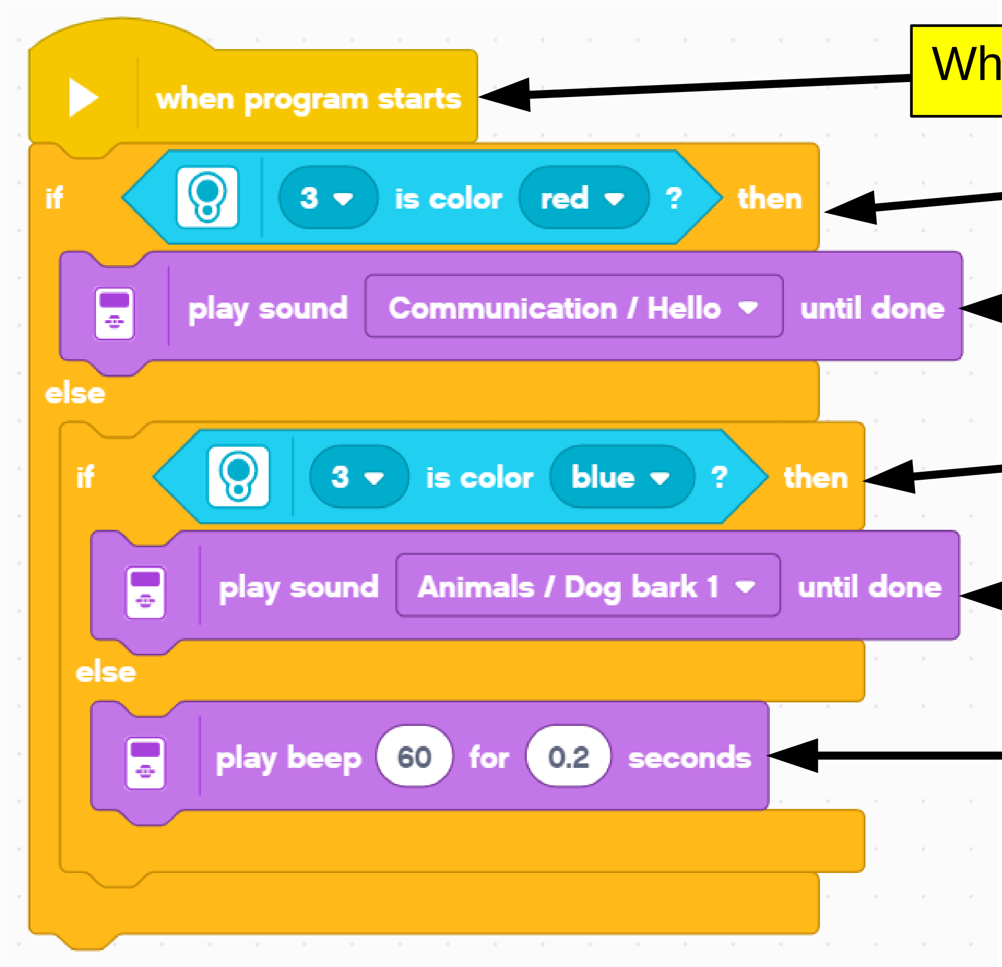
“else”, play beep

The Spike Prime will have a different set of sound. Use whatever sound you like!

Try it out!

- Add a color sensor to your robot
- Run the program on your robot
- Try different colors and sounds
- What did you observe?

Example 2



```
when program starts
if 3 is color red ? then
    play sound Communication / Hello until done
else
    if 3 is color blue ? then
        play sound Animals / Dog bark 1 until done
    else
        play beep 60 for 0.2 seconds
```

When the program starts...

Check if the color sensor is seeing red...

If it is, play “hello”...

Check if the color sensor is seeing blue...

If it is, play “Woof!”...

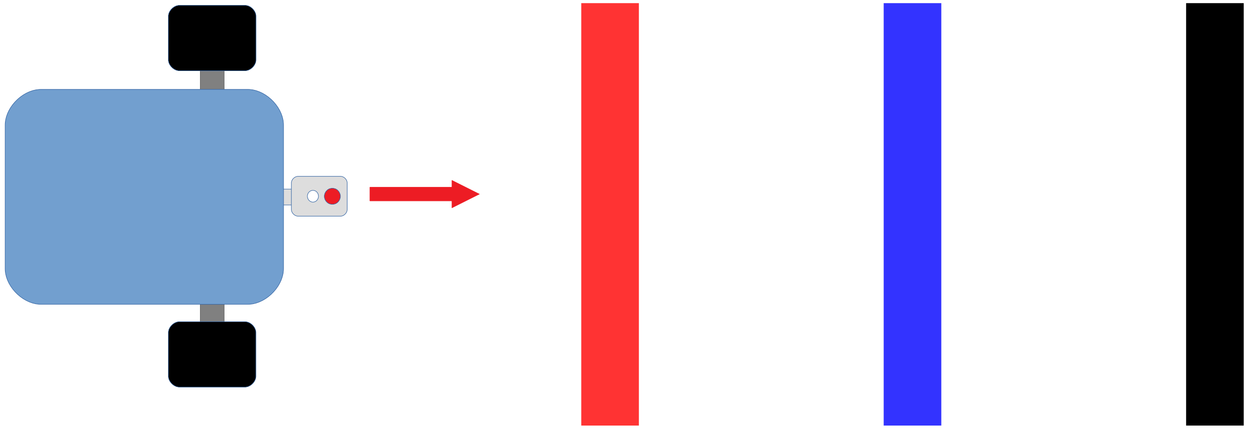
“else”, play beep

You can combine “if / else” to check for multiple conditions

Try it out!

A POSTERIORI
Play · Experience · Learn

Color Decoder



“Meow!”

“Woof!”

Robot Stops

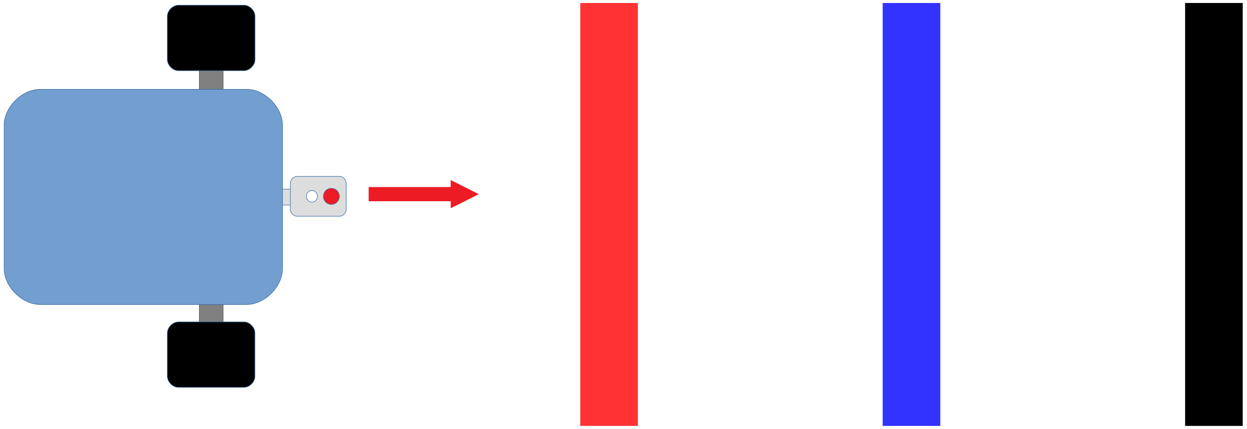
Note:

- The colors and the distance between each color may be randomized.
- You can choose whatever sound you like

A POSTERIORI

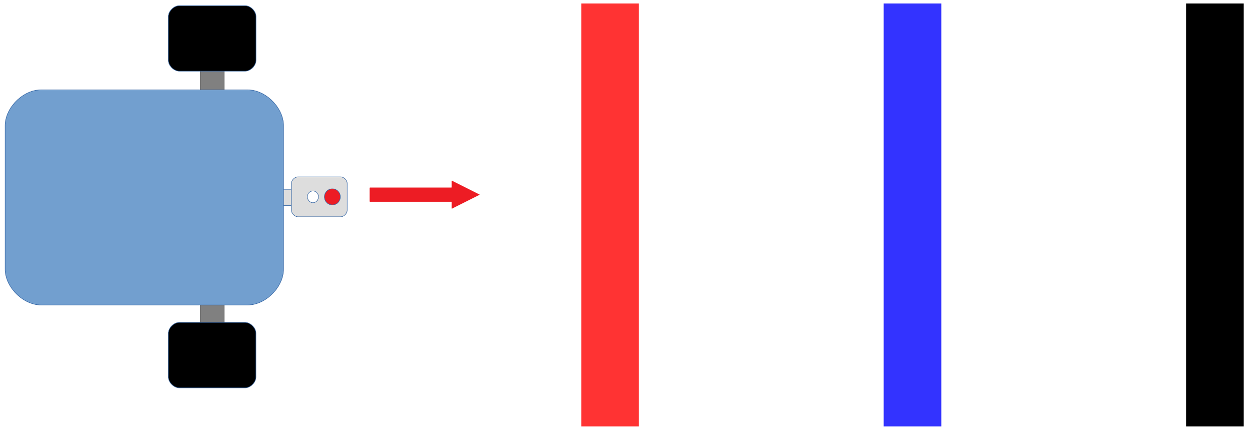
Play · Experience · Learn

What should the robot do?



- If it sees red, play “meow”
- If it sees blue, play “woof”
- If it sees black, stop
- Otherwise, it should move forward
- **Important**: It should keep repeating these!

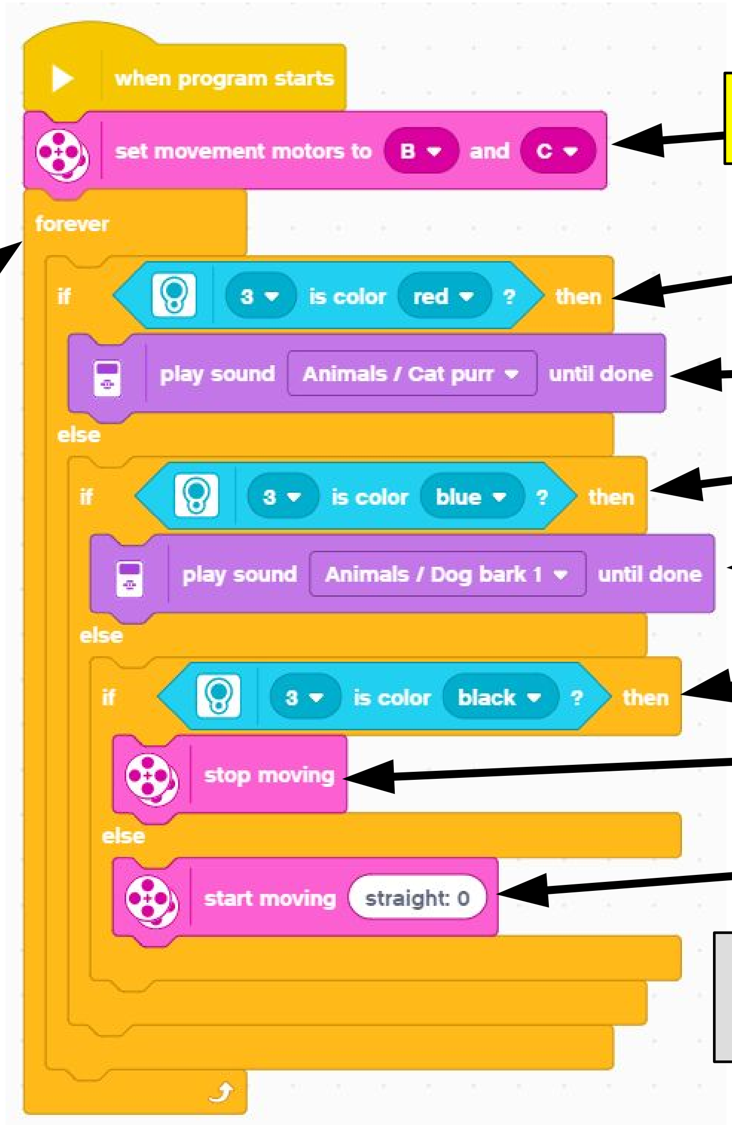
How to program?



- Let’s think about it...
  - Many conditions (eg. red, blue, black)
    - Need to use multiple “if / else”
  - Need to keep checking the color
    - Need to use a loop

A POSTERIORI
Play · Experience · Learn

Color Decoder



```
when program starts
set movement motors to B and C
forever
    if 3 is color red ? then
        play sound Animals / Cat purr until done
    else
        if 3 is color blue ? then
            play sound Animals / Dog bark 1 until done
        else
            if 3 is color black ? then
                stop moving
            else
                start moving straight: 0
```

Repeat Forever

Don’t forget to set the movement motors

Check if the color sensor is seeing red...

If it is, play “meow”...

Check if the color sensor is seeing blue...

If it is, play “Woof!”...

Check if the color sensor is seeing black...

If it is, stop moving

“else”, move forward

Notice we didn’t tell the robot how far to move. We just tell it to move straight.

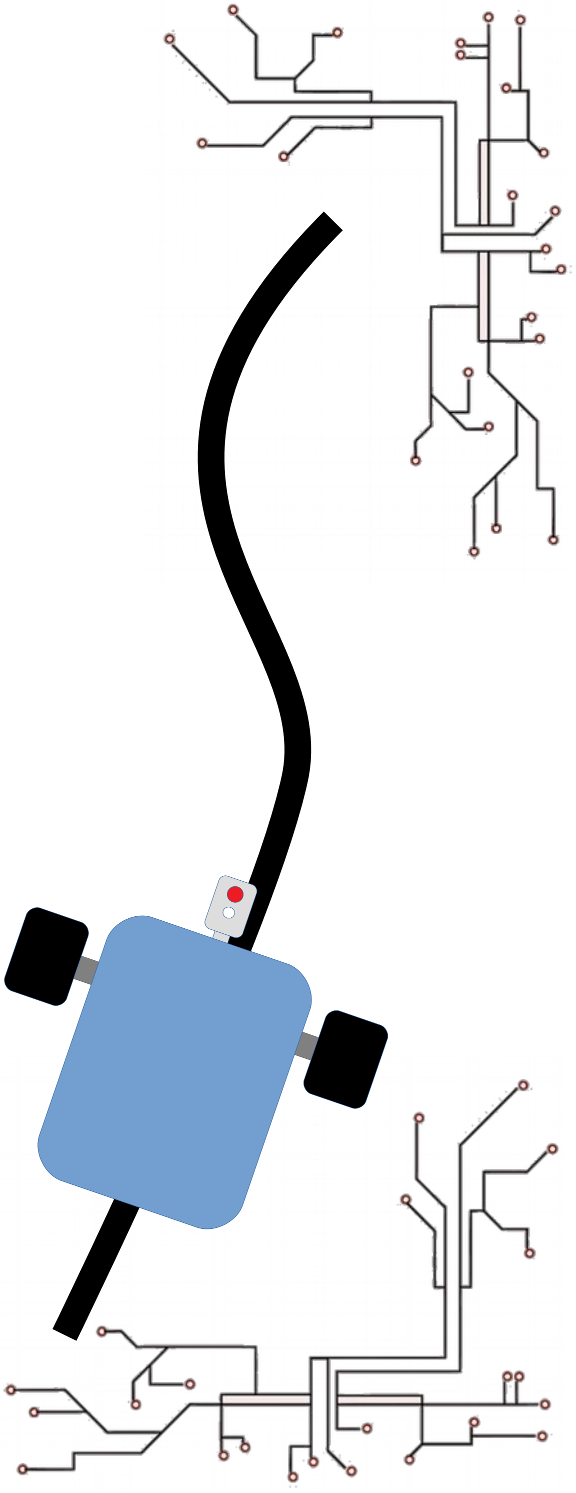
Try it out!

A POSTERIORI
Play · Experience · Learn

Line Following

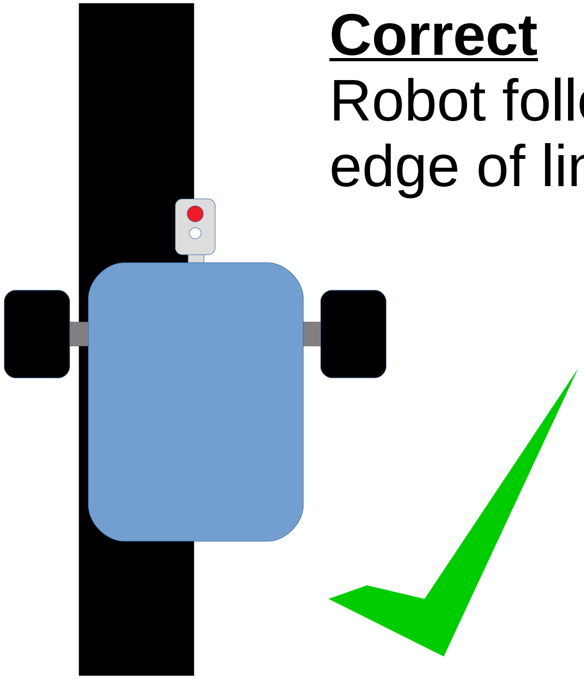
- Commonly used to guide robots
- Uses color sensor and “if / else”

Line following book drop robot in Tampines Library

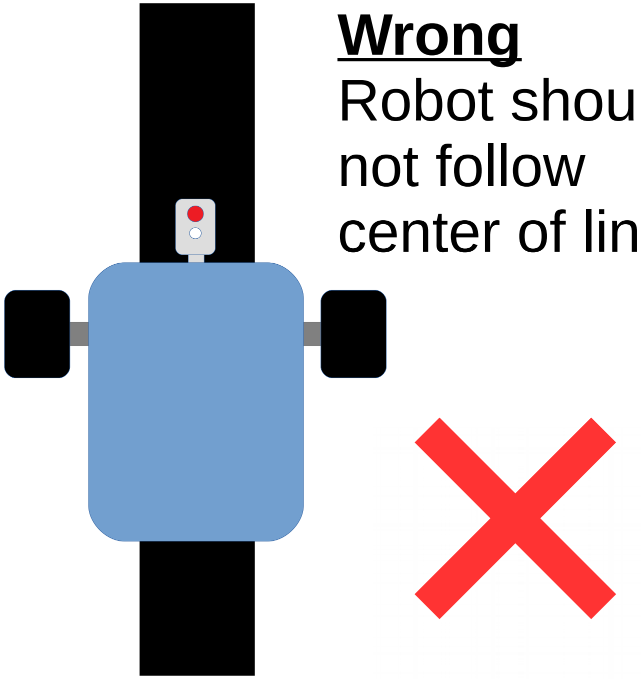


Line Following

- Trick: The robot doesn’t follow the center of the line, it follows the **edge** of the line



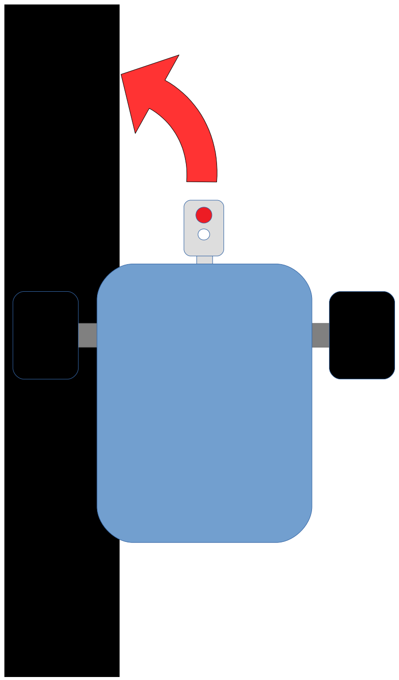
Correct
Robot follows
edge of line



Wrong
Robot should
not follow
center of line

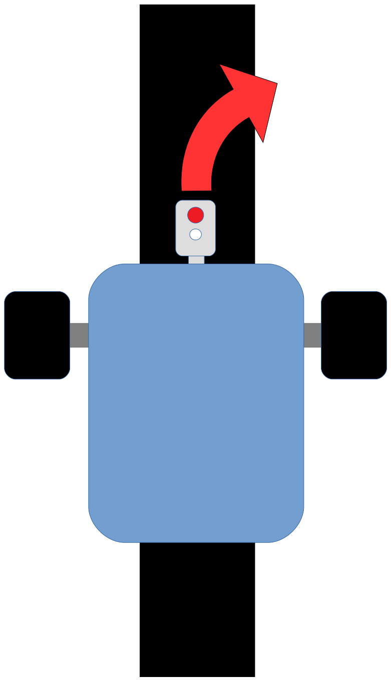
A POSTERIORI
Play · Experience · Learn

Line Following



If robot sees **white**, it is too far to the **right**...

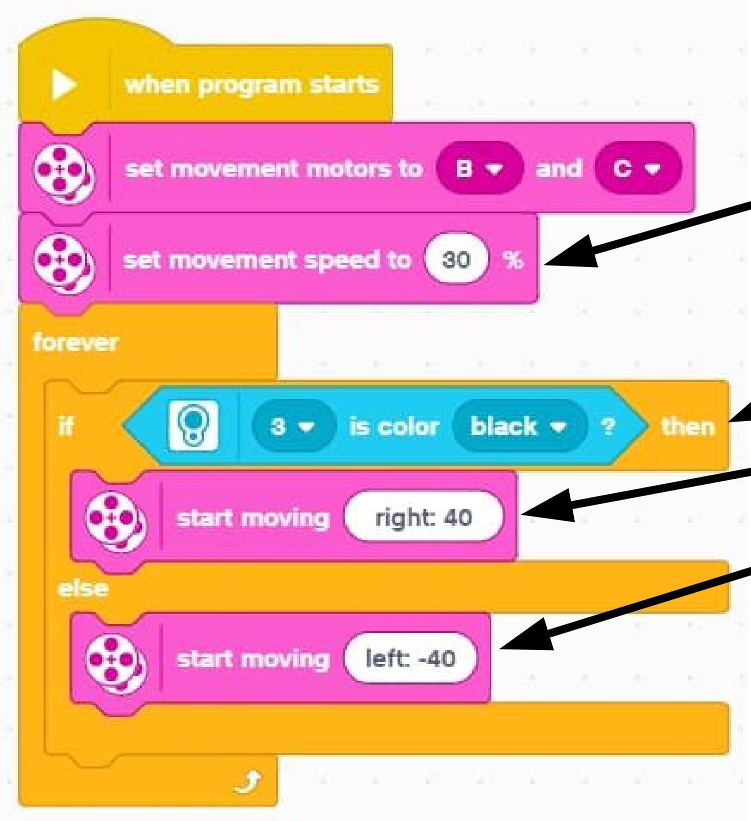
...so it should turn **left**.



If robot sees **black**, it is too far to the **left**...

...so it should turn **right**

2 States Line Follower



```
when program starts
set movement motors to B and C
set movement speed to 30 %
forever
    if 3 is color black ? then
        start moving right: 40
    else
        start moving left: -40
```

Optional: Set the speed to 30%

Check if the color sensor is seeing black...

If it is, move right

“else”, move left

This is called a “2-states line follower” program, because the robot only have 2 states; “move left” and “move right”

Try it out!

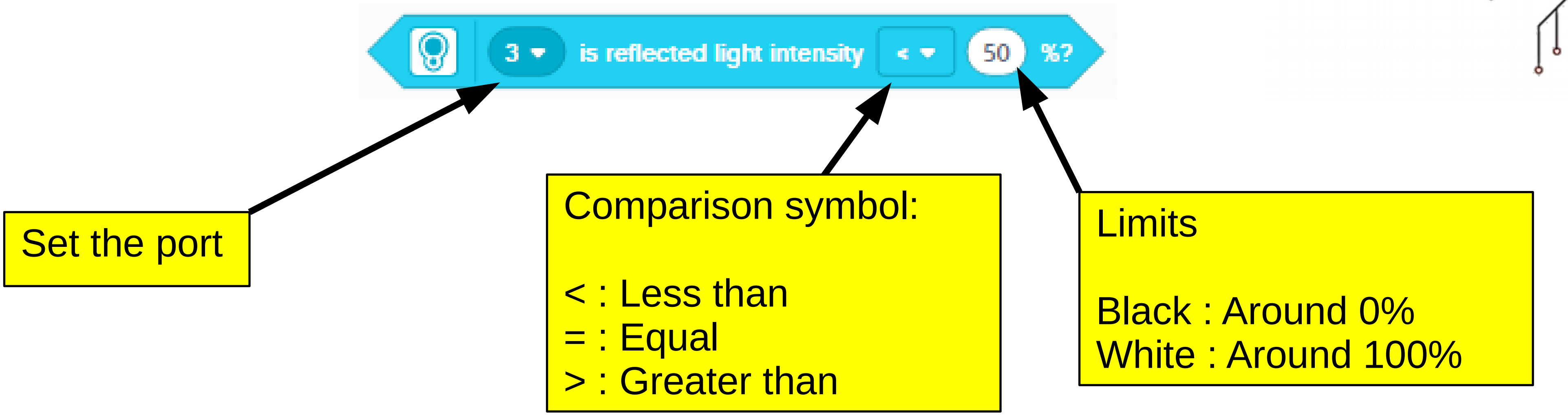
A POSTERIORI
Play · Experience · Learn

Problems

- Problem:
  - Movement is slow and jerky
- Why?:
  - Robot ONLY move left and right. It never goes straight.
- Can we have 3 states?
  - Black, White, and Gray? (...there’s no “Gray”)

Solution

Check the intensity of the reflected light!



Set the port

Comparison symbol:

< : Less than
= : Equal
> : Greater than

Limits

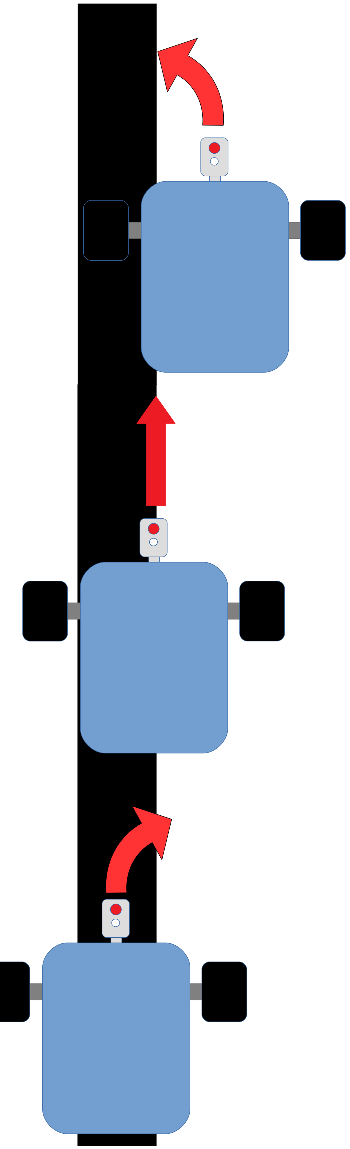
Black : Around 0%
White : Around 100%

Important
Black should be close to 0%, but won't be exactly 0%
White should be close to 100%, but won't be exactly 100%

You'll need to test it out to find the actual values

A POSTERIORI
Play · Experience · Learn

3 States Line Follower

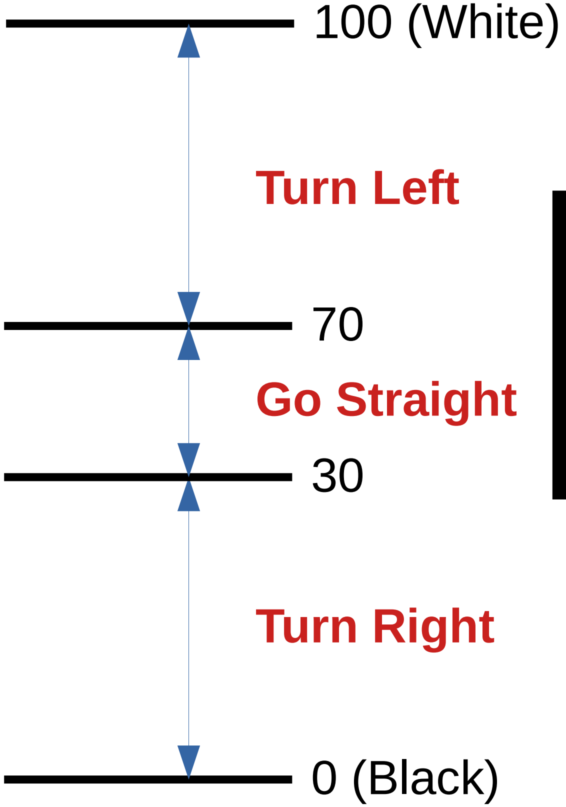


Robot sees white (> 70%):
Turn Left

Robot sees half white, half black (30% to 70%):
Go Straight

Robot sees black (< 30%):
Turn Right

Light Sensor Value

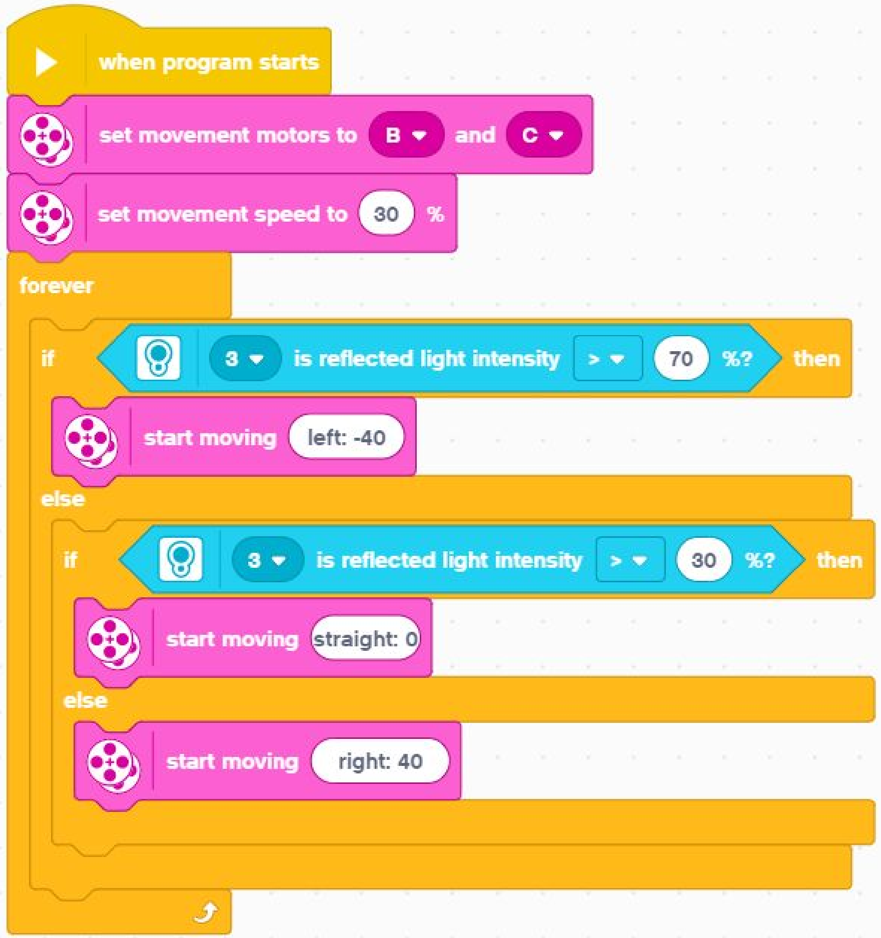


Pseudo Code

```
IF Value > 70:
  Turn Left
ELSE IF Value > 30:
  Go Straight
ELSE
  Turn Right
```

A POSTERIORI
Play · Experience · Learn

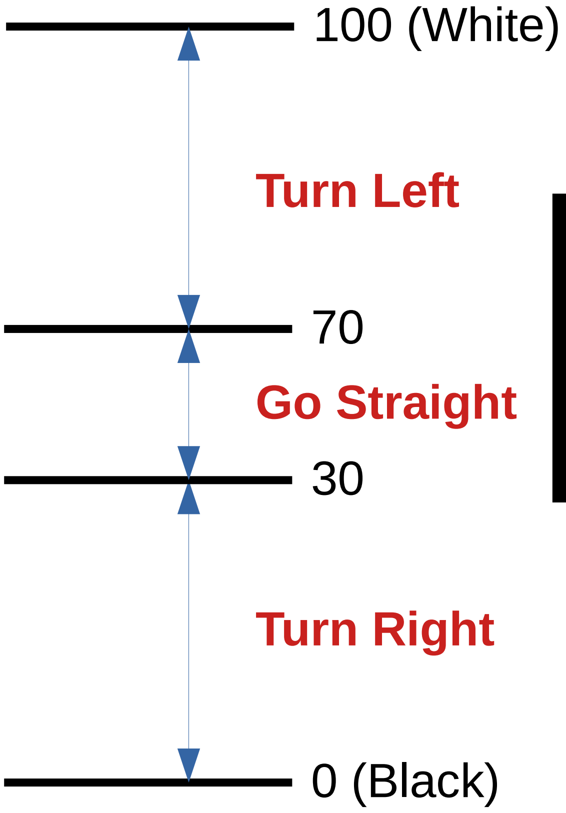
3 States Line Follower



```
when program starts
set movement motors to B and C
set movement speed to 30 %
forever
    if 3 is reflected light intensity > 70 %? then
        start moving left: -40
    else
        if 3 is reflected light intensity > 30 %? then
            start moving straight: 0
        else
            start moving right: 40
```

Try it out!

Light Sensor Value



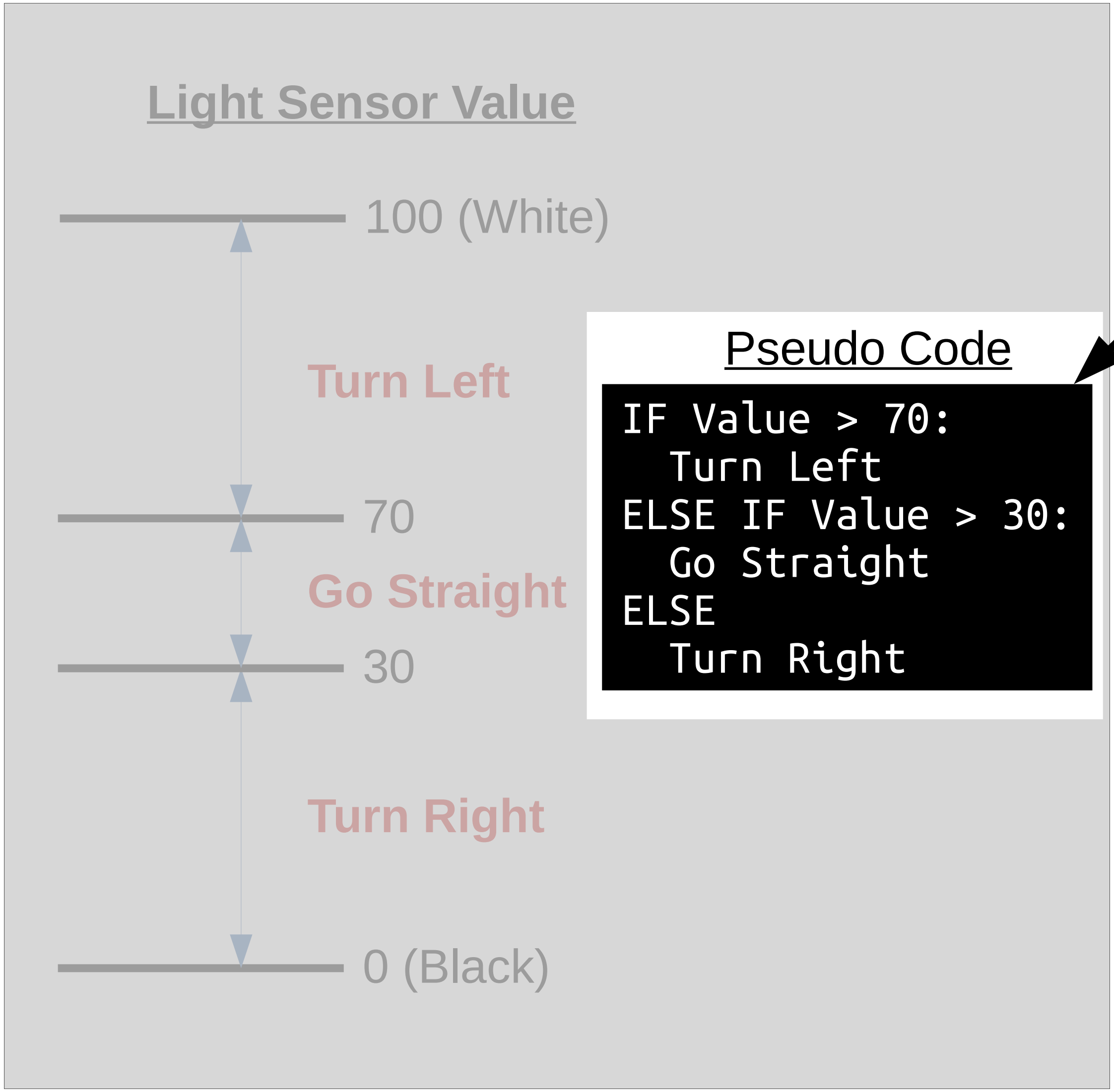
Pseudo Code

```
IF Value > 70:
  Turn Left
ELSE IF Value > 30:
  Go Straight
ELSE
  Turn Right
```

A POSTERIORI

Play · Experience · Learn

Pseudo Code



Pseudo Code

```
IF Value > 70:
  Turn Left
ELSE IF Value > 30:
  Go Straight
ELSE
  Turn Right
```

Pseudo Code
Looks like real programming code, but it won’t actually run.

Programmers use it to explain how to program without providing the exact details.

As we progress, we’ll start providing more examples in pseudo code without the blocks code.

A POSTERIORI
Play · Experience · Learn

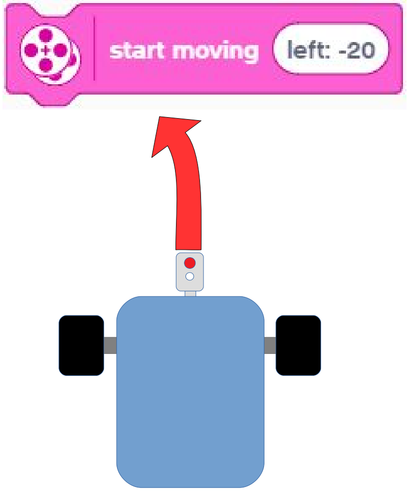
Can we do better?

- 2-states
  - Slow and Jerky
- 3-states
  - Smoother, but still a little jerky
- Can we have 5 states?

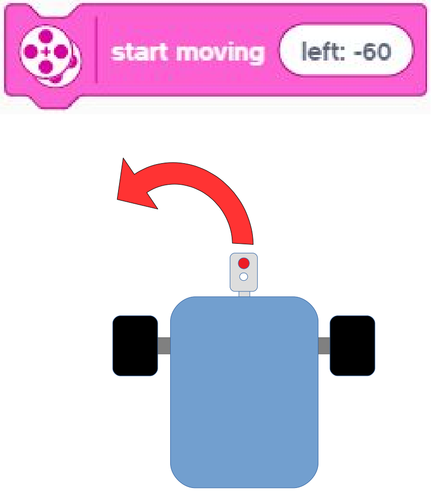
5 States Line Follower

Try it out!

- Read the pseudo code
- Try and write the actual program using that

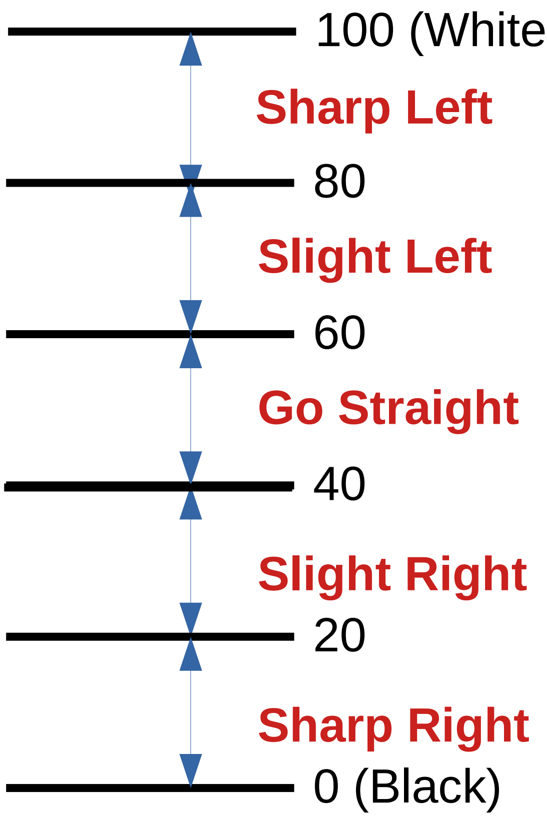


Slight Left



Sharp Left

Light Sensor Value



Pseudo Code

```
IF Value > 80:
  Turn Sharp Left
ELSE IF Value > 60:
  Turn Slight Left
ELSE IF Value > 40:
  Go Straight
ELSE IF Value > 20:
  Turn Slight Right
ELSE
  Turn Right
```

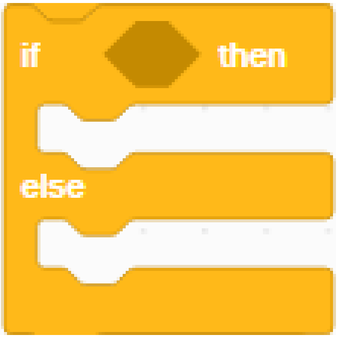
A POSTERIORI
Play · Experience · Learn

Can we do even better?

- You can make...
  - 7 states, 9 states, etc
  - ...but it’s probably not necessary
- More advanced method:
  - Proportional control
  - Uses math equation instead of “if / else”
  - Better than 5 states in some cases, but not all
  - We’ll leave that for a future lesson...

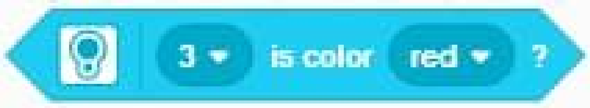
Summary

- Use “if / else” to choose what to do



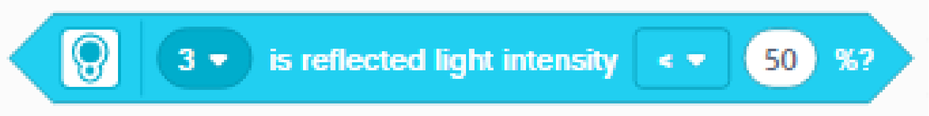
if then
else

- “is color” block to check color



3 is color red ?

- “reflected light intensity” block to check brightness

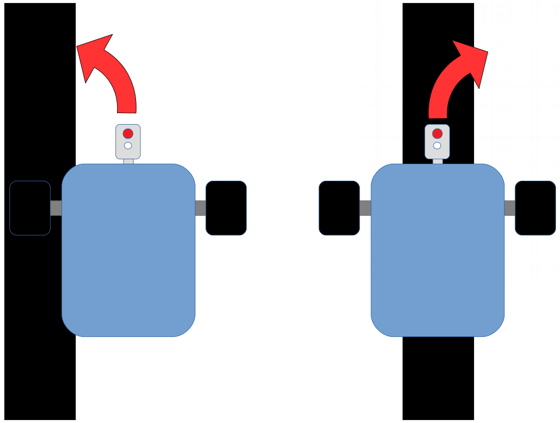


3 is reflected light intensity < 50 %?

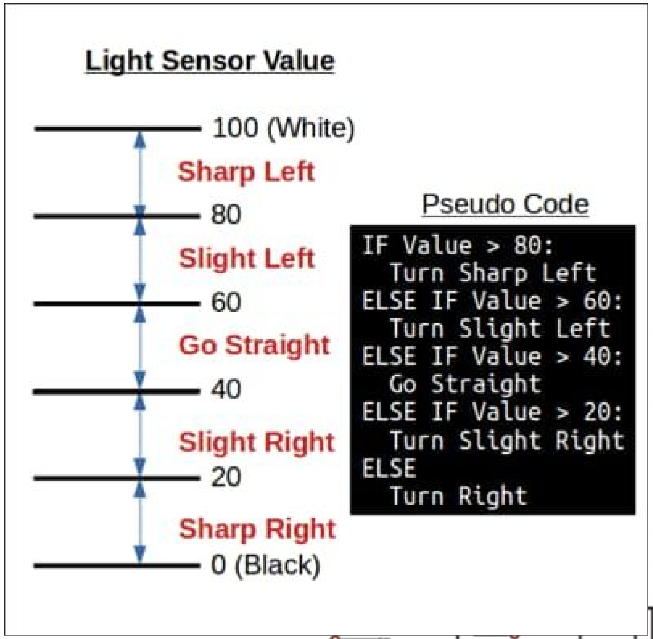
A POSTERIORI
Play · Experience · Learn

Summary

- Line Following:
  Robot follows edge of line



- Line Following:
  Can use more states to make movement smoother



Pseudo Code

```
IF Value > 80:
  Turn Sharp Left
ELSE IF Value > 60:
  Turn Slight Left
ELSE IF Value > 40:
  Go Straight
ELSE IF Value > 20:
  Turn Slight Right
ELSE
  Turn Right
```

Copyright

A POSTERIORI

Play · Experience · Learn