MOVING STRAIGHT

ARVIND SESHAN
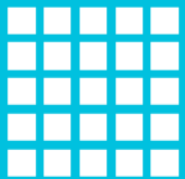
LESSON OBJECTIVES

1. Learn how to make your robot go forward and backwards

2. Learn how to use the Move functions
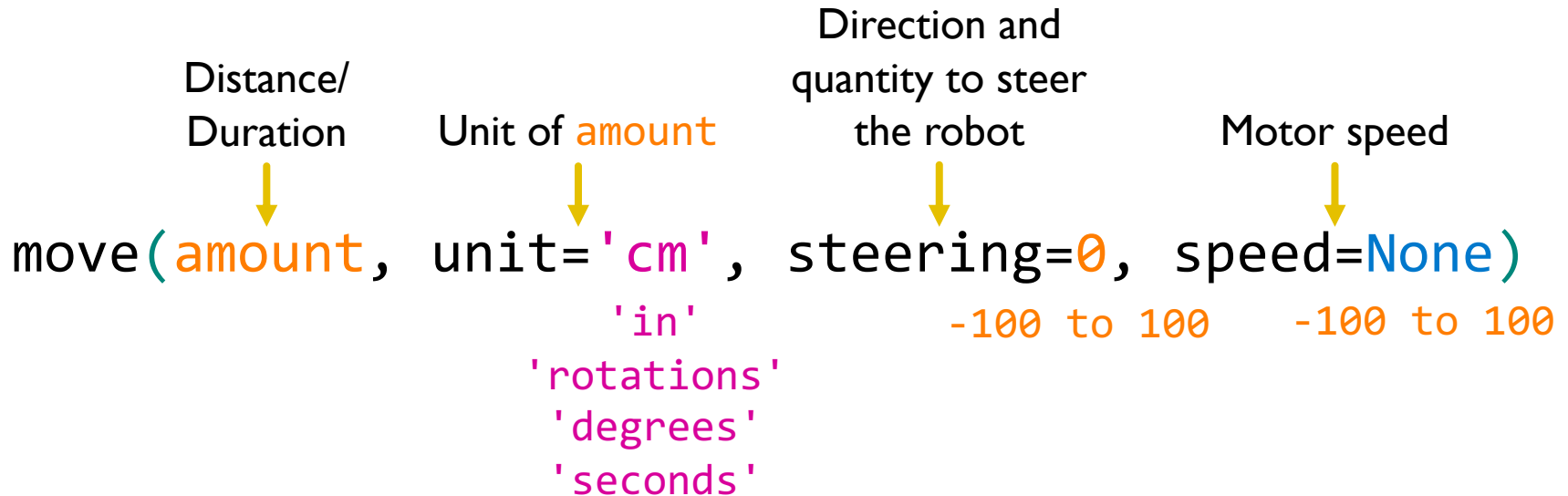
MOVE()

Distance/
Duration

Unit of amount

Direction and
quantity to steer
the robot

Motor speed

```
move(amount, unit='cm', steering=0, speed=None)
```

'in'
'rotations'
'degrees'
'seconds'

-100 to 100   -100 to 100

unit='cm', steering=0, and speed=None, are the default values if nothing is set. When speed=None, the speed value used is the default speed set by set_default_speed().

Set in Configuration
To use this function you will set the speed, stop mode, motor ports, wheel size (see Configuring Robot Movement Lesson)

MOVE_TANK()

Distance/
Duration

Unit of
amount

Left motor speed

Right motor speed

```
move_tank(amount, unit='cm', left_speed=None, right_speed=None)
                       'in'              -100 to 100        -100 to 100
                       rotations'
                       'degrees'
                       'seconds'
```

unit='cm', left_speed=None, and right_speed=None, are the default values if nothing is set. When left_speed=None and/or right_speed=None, the speed value used is the default speed set by set_default_speed().

Set in Configuration
To use this function you will set the speed, stop mode, motor ports, wheel size (see Configuring Robot Movement Lesson)
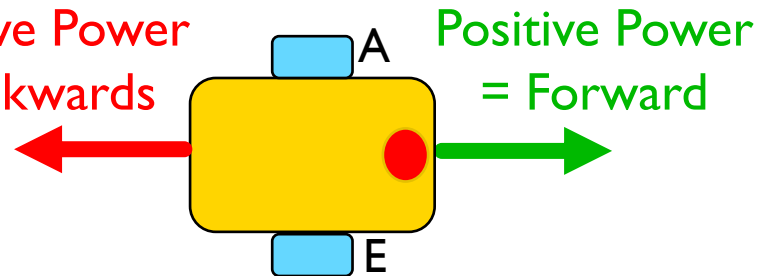
NEGATIVE VALUES

- You can enter negative values for power or distance

- This will make the robot move backwards

- If you negate two values (e.g. speed and distance negative), the robot will move forward.

Negative Power = Backwards

Positive Power = Forward

Negative Power = Backwards

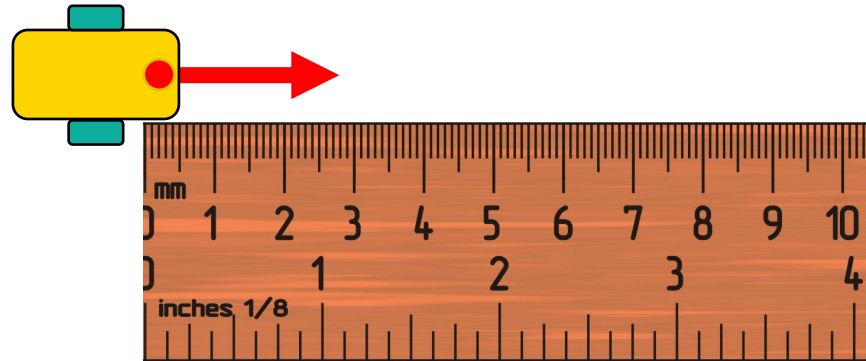Positive Power = Forward

A

E
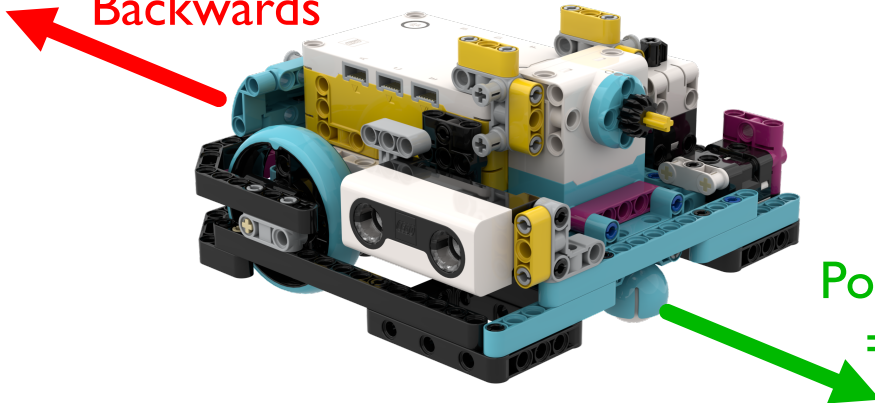
CHALLENGE 1: MOVE 10 CM

- Move the robot 10 centimeters forward

- Basic steps:

  - Configure your robot

  - Use a MotorPairs function (move() or move_tank()) and move forward for 10cm
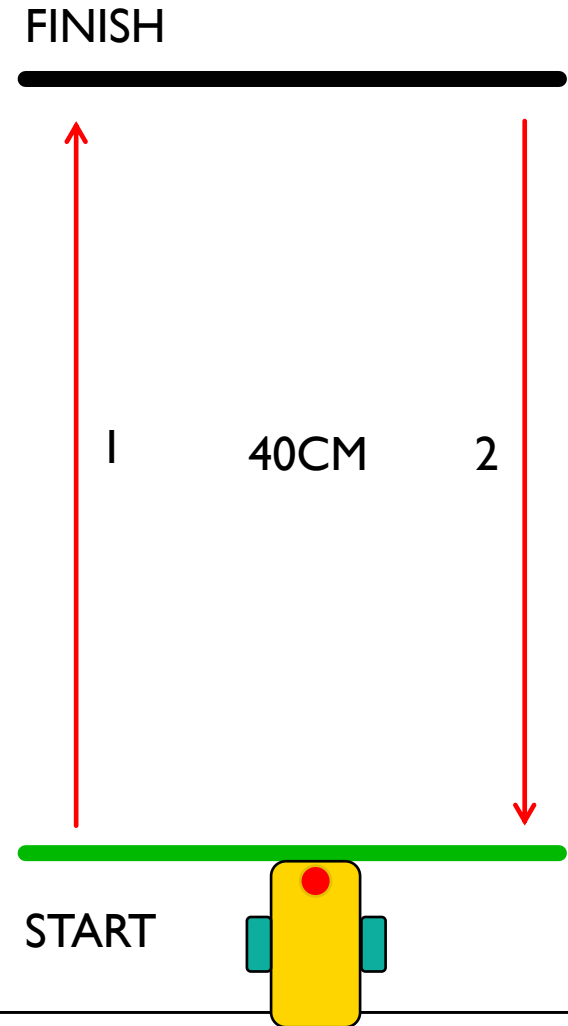
CHALLENGE I SOLUTION

■ Configure your robot

■ If you are using the smaller SPIKE Prime wheels on Droid Bot IV, set the one rotation to 17.5cm (in the code shown)

■ If you are using the larger SPIKE Prime wheels on ADB, remember to set one rotation to 27.6cm

■ Move forward for 10 cm. The same cm mode is available in the move_tank() function as well.

```
motor_pair = MotorPair('A', 'E')
motor_pair.set_stop_action('brake')
motor_pair.set_motor_rotation(17.5, 'cm')
motor_pair.set_default_speed(50)
motor_pair.move(15, 'cm')
```

CHALLENGE II: MOVE FORWARD AND BACK

■ Move your robot forward from the start line to the finish line (1) and back to the start (2)

■ Basic steps:

   ■ Configure your robot

   ■ Use a MotorPair function and move forward for the desired amount (40cm)

   ■ Use the same MotorPair function to move backwards (40cm)

FINISH

1   40CM   2

START

CHALLENGE II SOLUTION

- Configure your robot

- If you are using the smaller SPIKE Prime wheels on Droid Bot IV, set the one rotation to 17.5cm (in the code shown)

- If you are using the larger SPIKE Prime wheels on ADB, you will set one rotation to 27.6cm

- Robot moves forward 40cm and backwards 40cm by setting the distance to -40.

```
motor_pair = MotorPair('A', 'E')
motor_pair.set_stop_action('brake')
motor_pair.set_motor_rotation(17.5, 'cm')
motor_pair.set_default_speed(50)
motor_pair.move(40, 'cm')
motor_pair.move(-40, 'cm')
```

START MOVING AND STOP MOVING FUNCTIONS

- There are 5 more move functions

- The start functions will turn **on** your drive motors at the given speed (and steering if given).

- These functions have no duration/distance. After turning the motor on, the program instantly moves to the next line

- The motor will continue running until stopped or controlled by another function

- stop() will halt your drive motors no matter what action they are running.

- There are also functions that allow you to control motor power instead of speed.

```
start(steering=0, speed=None)
stop()
start_tank(left_speed, right_speed)
start_at_power(power, steering=0)
start_tank_at_power(left_power, right_power)
```

WAIT FUNCTIONS AND CHALLENGE III

- Since Start and Stop Moving functions execute instantly, they need to be used with other functions to be made useful. One common way they are used is with Wait Functions. Wait Functions hold up the program execution until some event occurs. The lessons on sensors cover Wait Functions in more detail.

- For now, we will use wait_for_seconds()

```
wait_for_seconds(seconds)
```

- This functions takes the entered number of seconds to run

Challenge III:
Use Start Moving, Stop Moving and Wait Functions to make the robot move forward for 3 seconds

CHALLENGE III: MOVING FOR 3 SECONDS

■ Can you Move 3 Seconds using just the Start Moving and Wait blocks?

```
motor_pair = MotorPair('A', 'E')
motor_pair.set_stop_action('brake')
motor_pair.start()
wait_for_seconds(3)
motor_pair.stop()
```

■ The start() function starts the robot moving

■ After turning on the motors, the program begins running the wait_for_seconds() functions. This takes 3 seconds to run.

■ The stop() function makes the robot stop

CREDITS

- This lesson was created by Arvind Seshan for SPIKE Prime Lessons

- More lessons are available at www.primelessons.org