



# Armstrong

## School Program 2023-2024

### Orientation Session



# Armstrong

entertainment meets education

REGISTER NOW



<https://armstrongedu.com/>



# Lesson Content



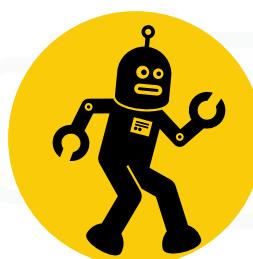
**Revising motor movement**



**Turning the robot**



**Functions**



**Moving the robot**



# Remember

## Setting PWM.

```
void setup() {  
    pinMode(5, OUTPUT);  
}
```

PWM pins 3, 5, 6, 9, 10, 11.

```
void loop() {  
    analogWrite(3, 0);  
}
```

number between 0 and 255



# Remember

Code to move robot forward with speed

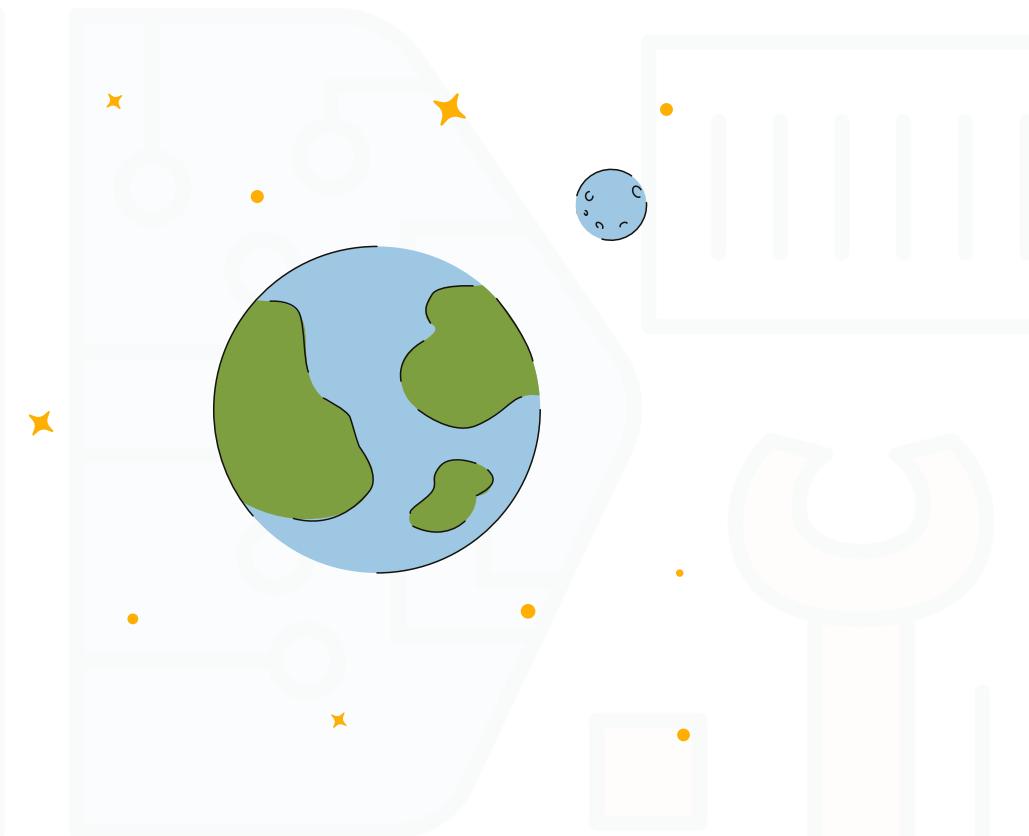
# Rotation

**What is rotation?**

It is a circular motion about an axis or center and this center is called axis of rotation.

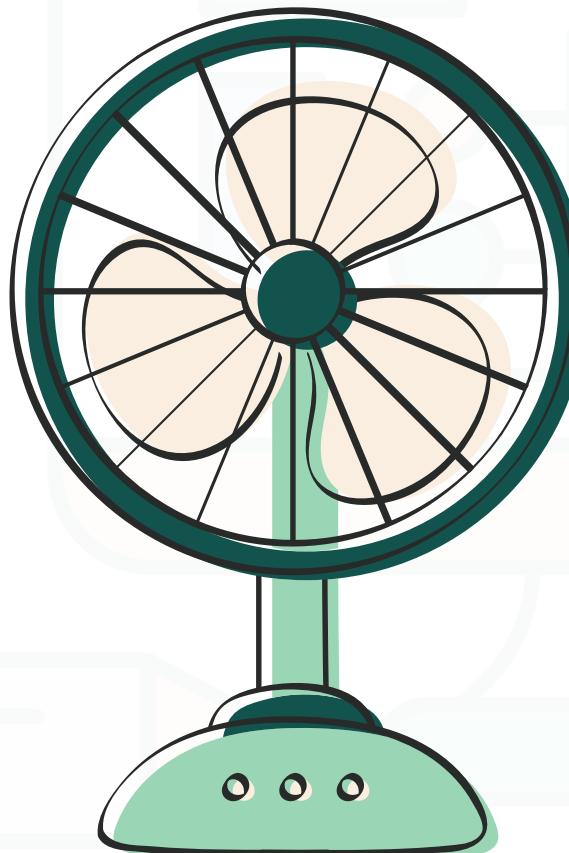
**How do we rotate?**

Depending on the axis of rotation a force is applied to the moving body.

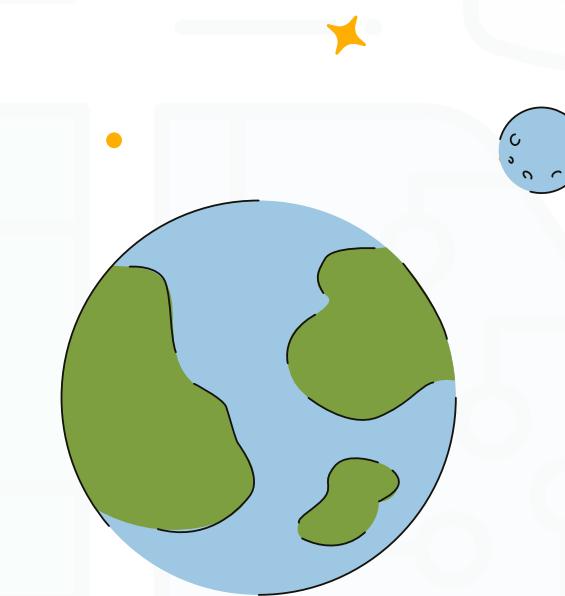


# Axis of Rotation

Object's Center

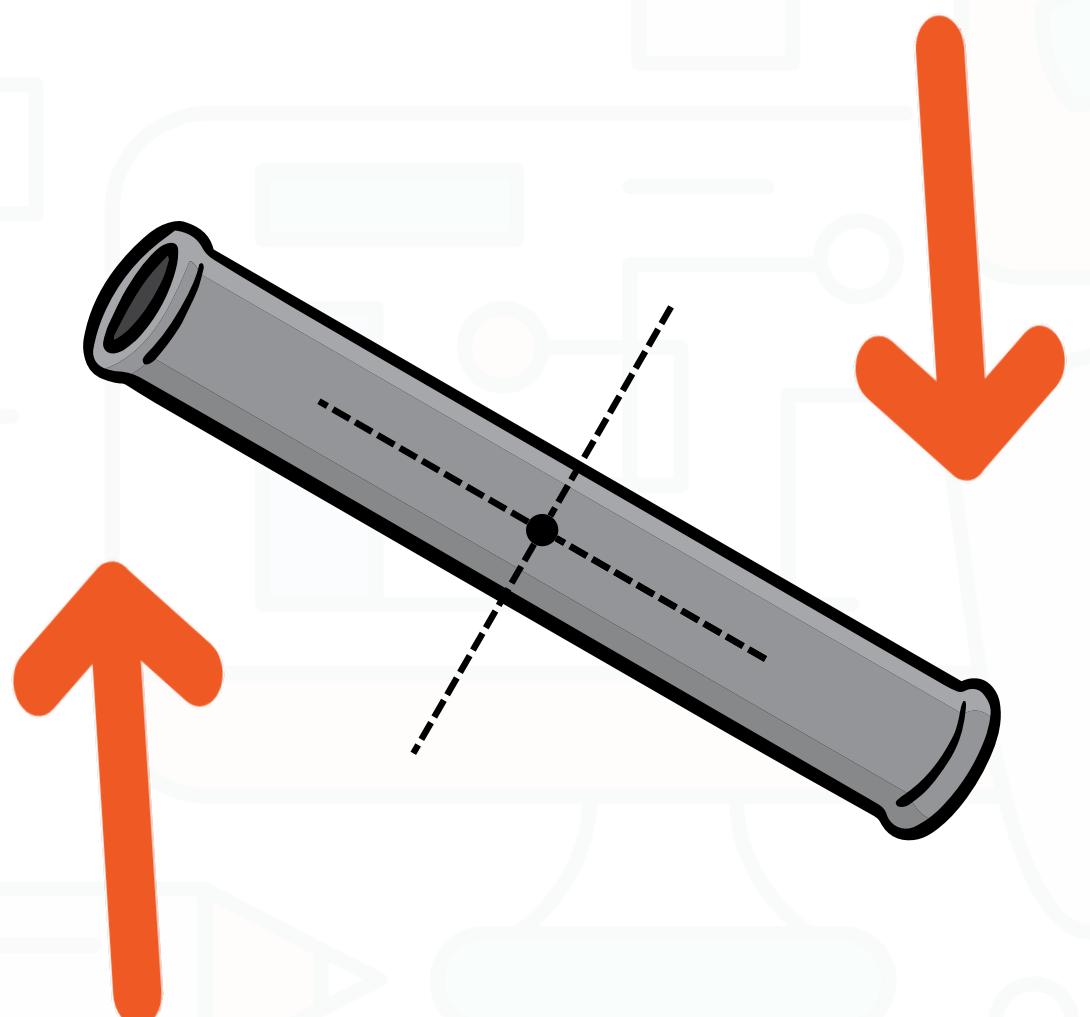


Point Outside The  
Objects Body

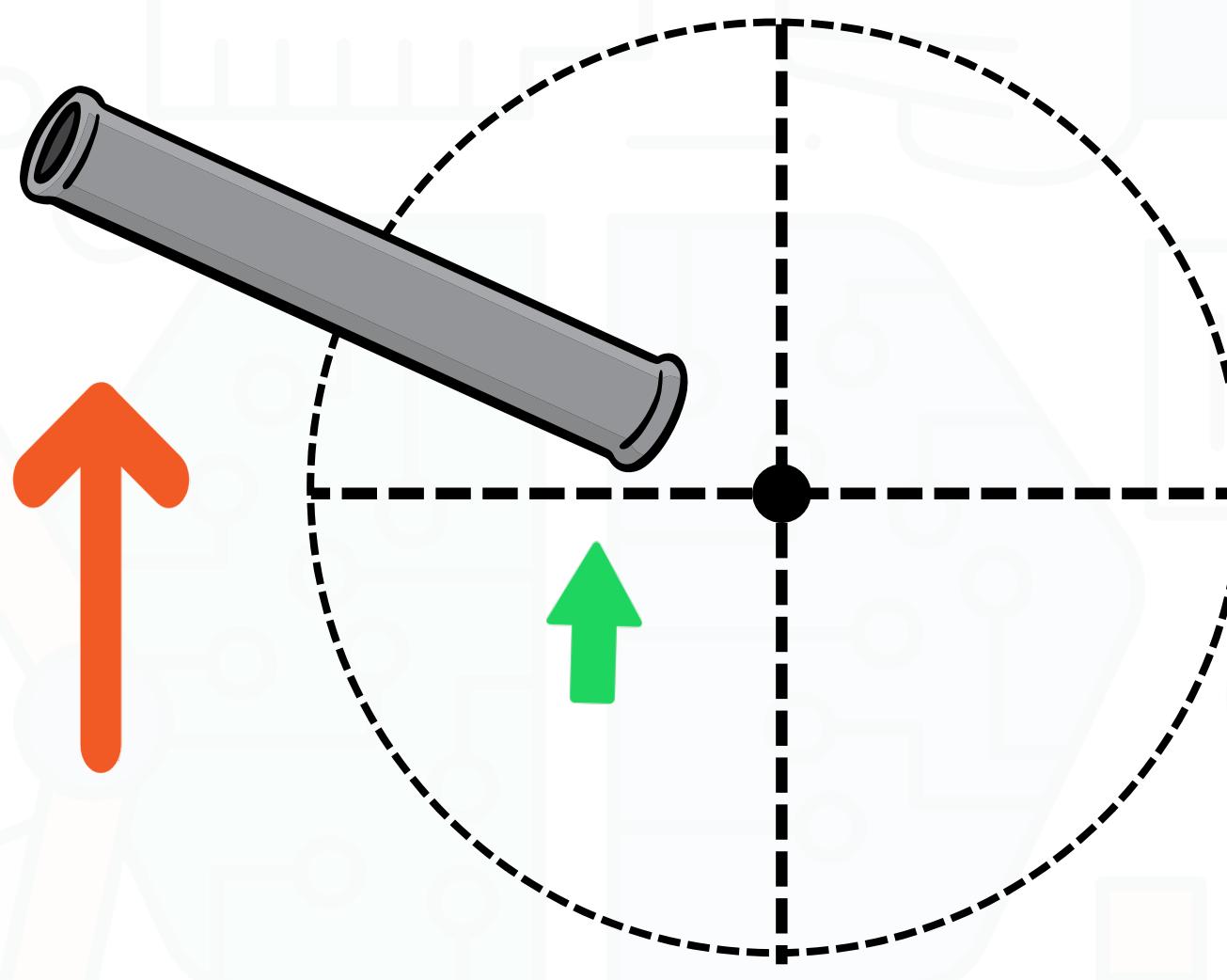


# Axis of Rotation

Object's Center



Point Outside The  
Objects Body



# Think

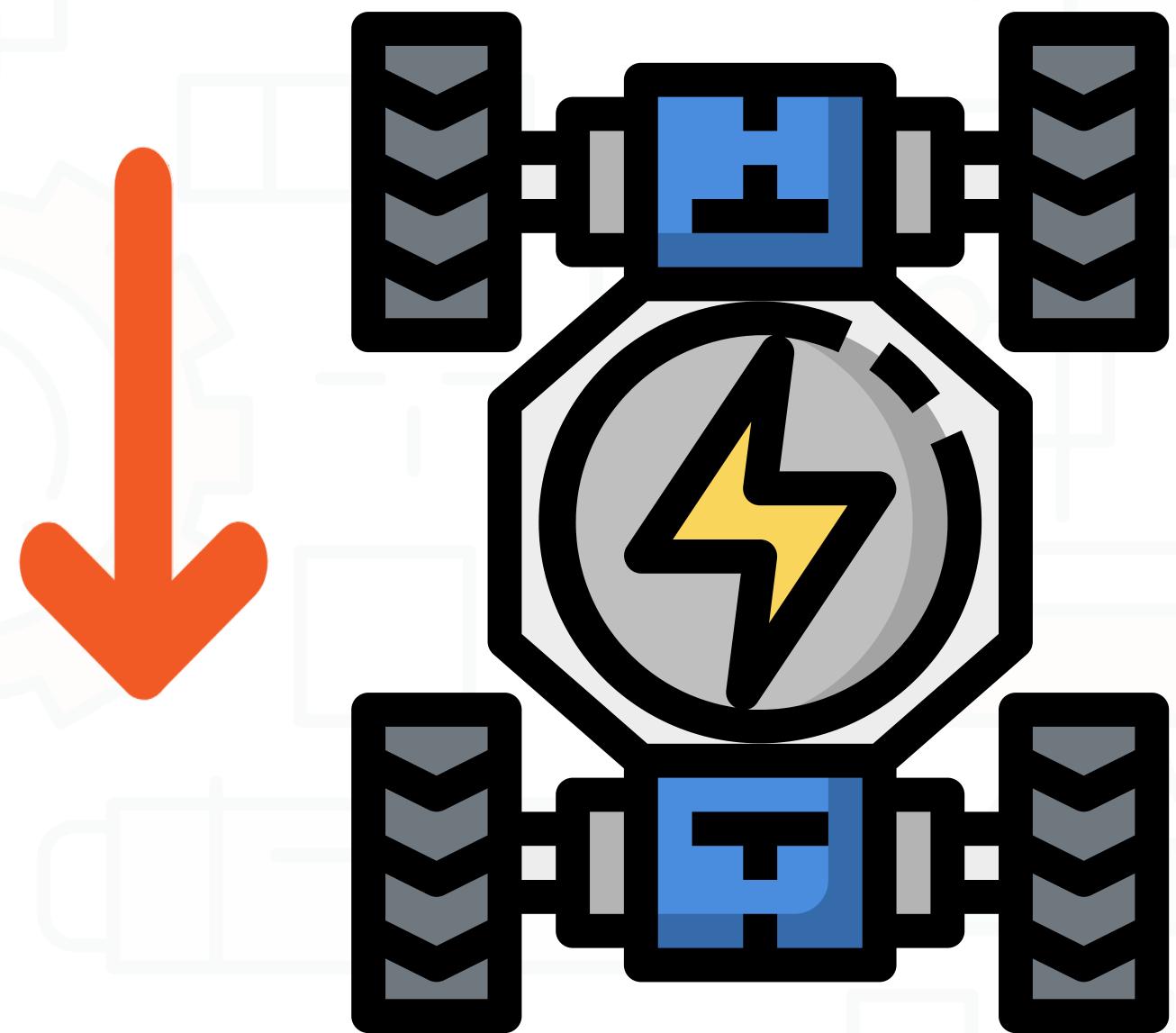
How can we make the robot rotate about its center?



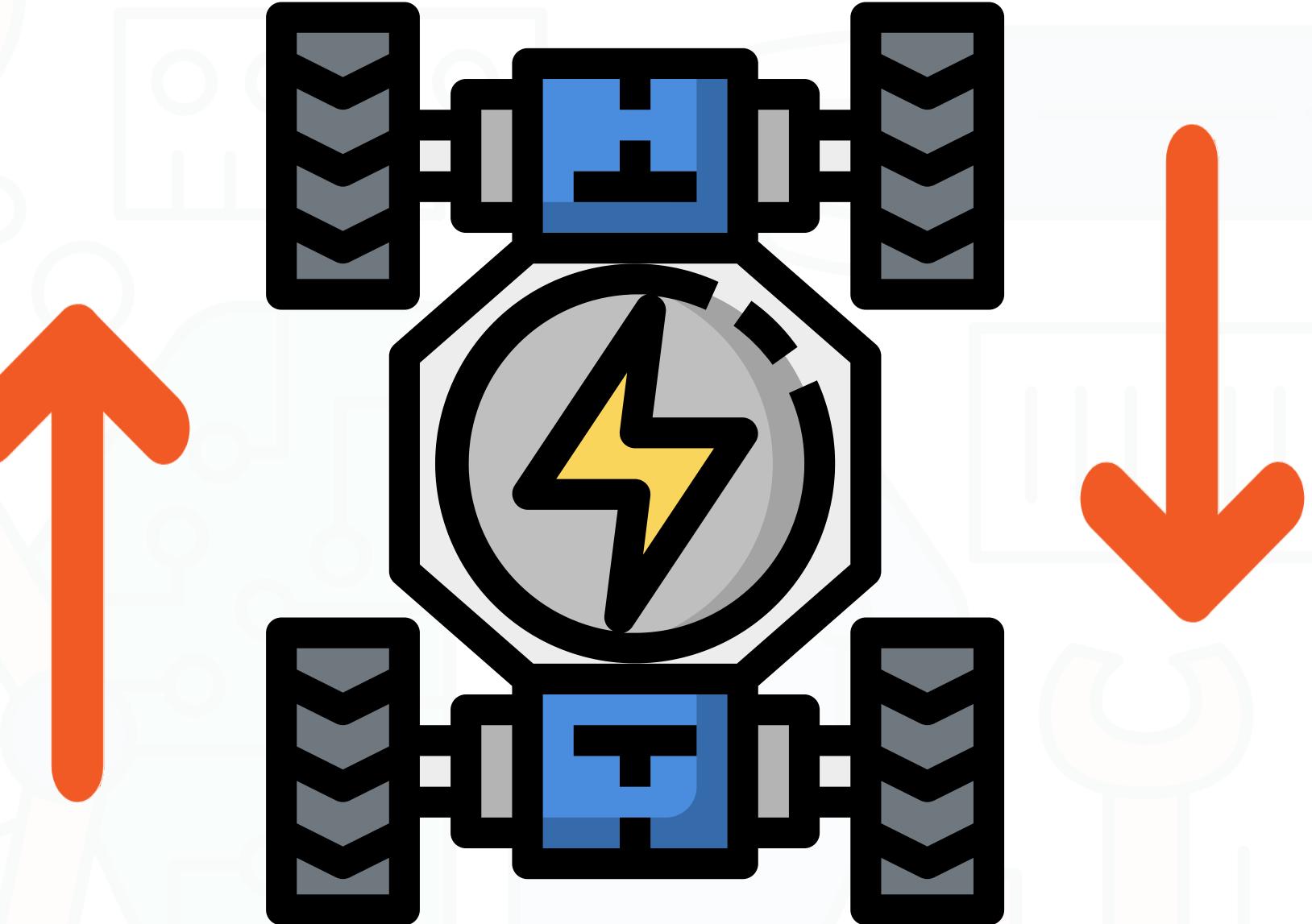
# Rotate The Robot



Left Rotation



Right Rotation

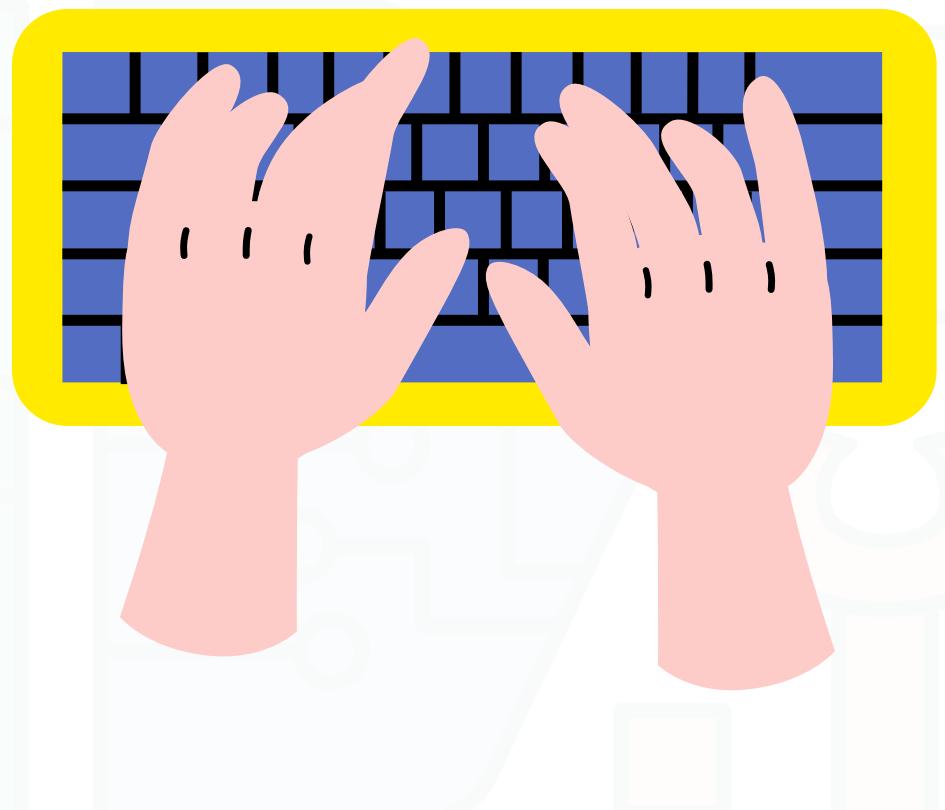


# Let's code

**Write a code to rotate the robot about its center.**

**Try it by yourself**

**Hint:** Remember forward and backward codes.

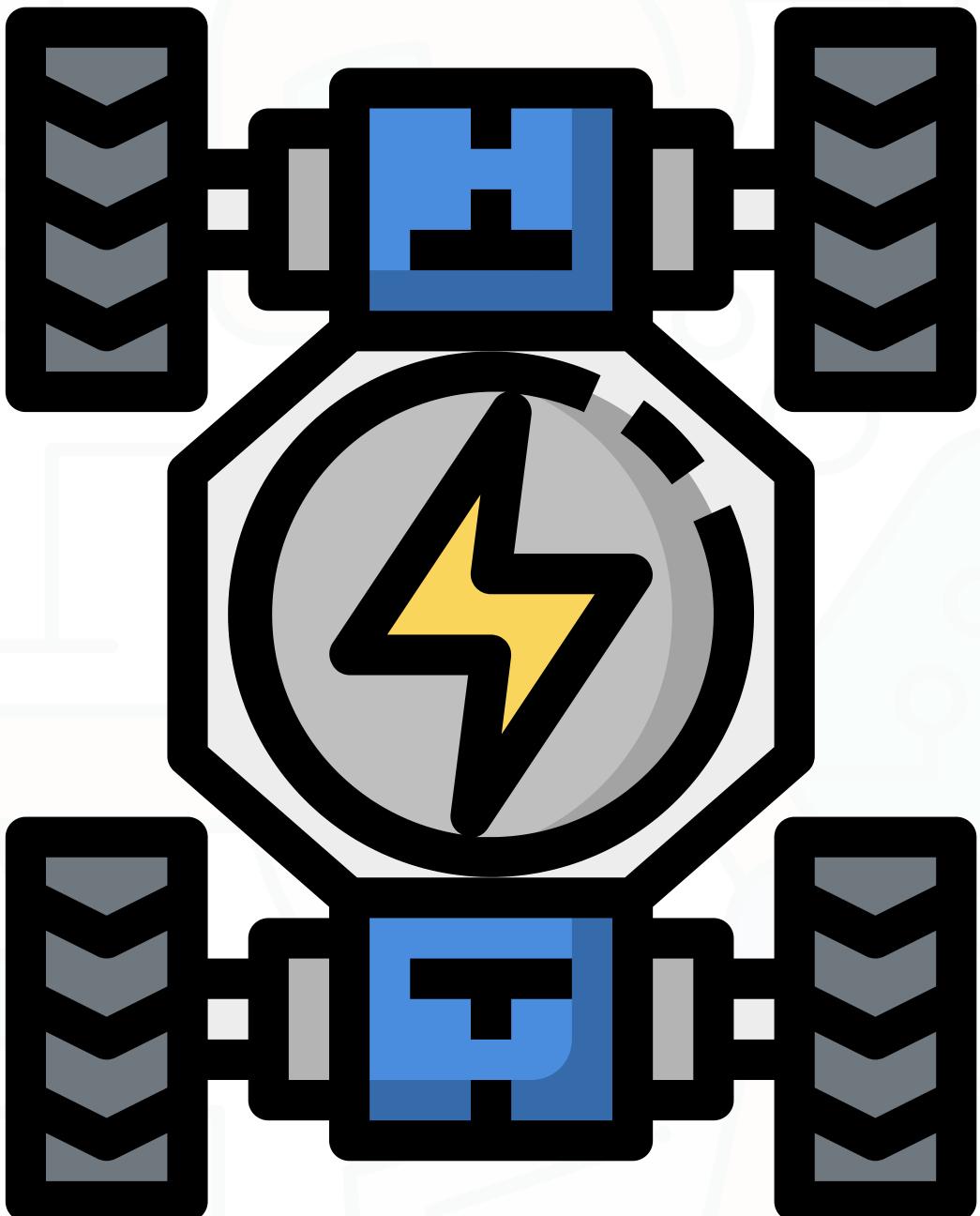


# Rotate The Robot



## Left Rotation

- Enable -> D6
- Input 1 -> D11
- Input2 -> D12



- Enable -> D5
- Input 1 -> D7
- Input2 -> D8

- Enable -> D10
- Input 1 -> A1(D15)
- Input2 -> A2(D16)

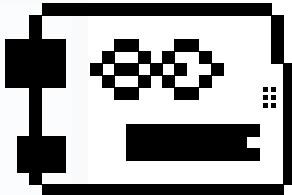
- Enable -> D9
- Input 1 -> D13
- Input2 -> A0(D14)

# Let's rotate the robot



## Step 1: Set motors' pwm modes and speed

```
//define motors PWM pins  
pinMode(5, OUTPUT); //back left motor PWM pin  
pinMode(6, OUTPUT); //front left motor PWM pin  
pinMode(9, OUTPUT); //back right motor PWM pin  
pinMode(10, OUTPUT); //front right motor PWM pin  
  
//setting motors speeds to 255  
  
analogWrite(5, 255);  
analogWrite(6, 255);  
analogWrite(9, 255);  
analogWrite(10, 255);
```

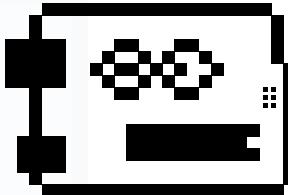


# Let's rotate the robot



## Step 1: Set motors' direction pins' modes

```
//defining motors directions pin  
pinMode(7, OUTPUT); //back left motor pin 1  
pinMode(8, OUTPUT); //back left motor pin 2  
  
pinMode(11, OUTPUT); //front left motor pin 1  
pinMode(12, OUTPUT); //front left motor pin 2  
  
pinMode(13, OUTPUT); //back right motor pin 1  
pinMode(A0, OUTPUT); //back right motor pin 2  
  
pinMode(A1, OUTPUT); //front right motor pin 1  
pinMode(A2, OUTPUT); //front right motor pin 2
```



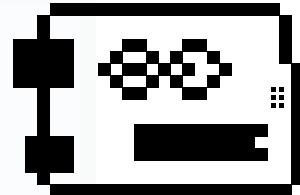
# Let's rotate the robot



## Step 2: Move the right wheels forward

```
//Back right motor forward  
digitalWrite(13, HIGH);  
digitalWrite(A0, LOW);
```

```
//front right motor forward  
digitalWrite(A1, HIGH);  
digitalWrite(A2, LOW);
```



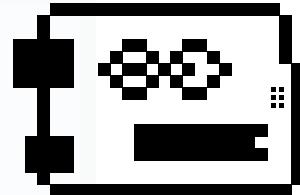
# Let's rotate the robot



## Step 3: Move the left wheels backward

```
//front left motor backward  
digitalWrite(11, LOW);  
digitalWrite(12, HIGH);
```

```
//back left motor backward  
digitalWrite(7, LOW);  
digitalWrite(8, HIGH);
```



# Think

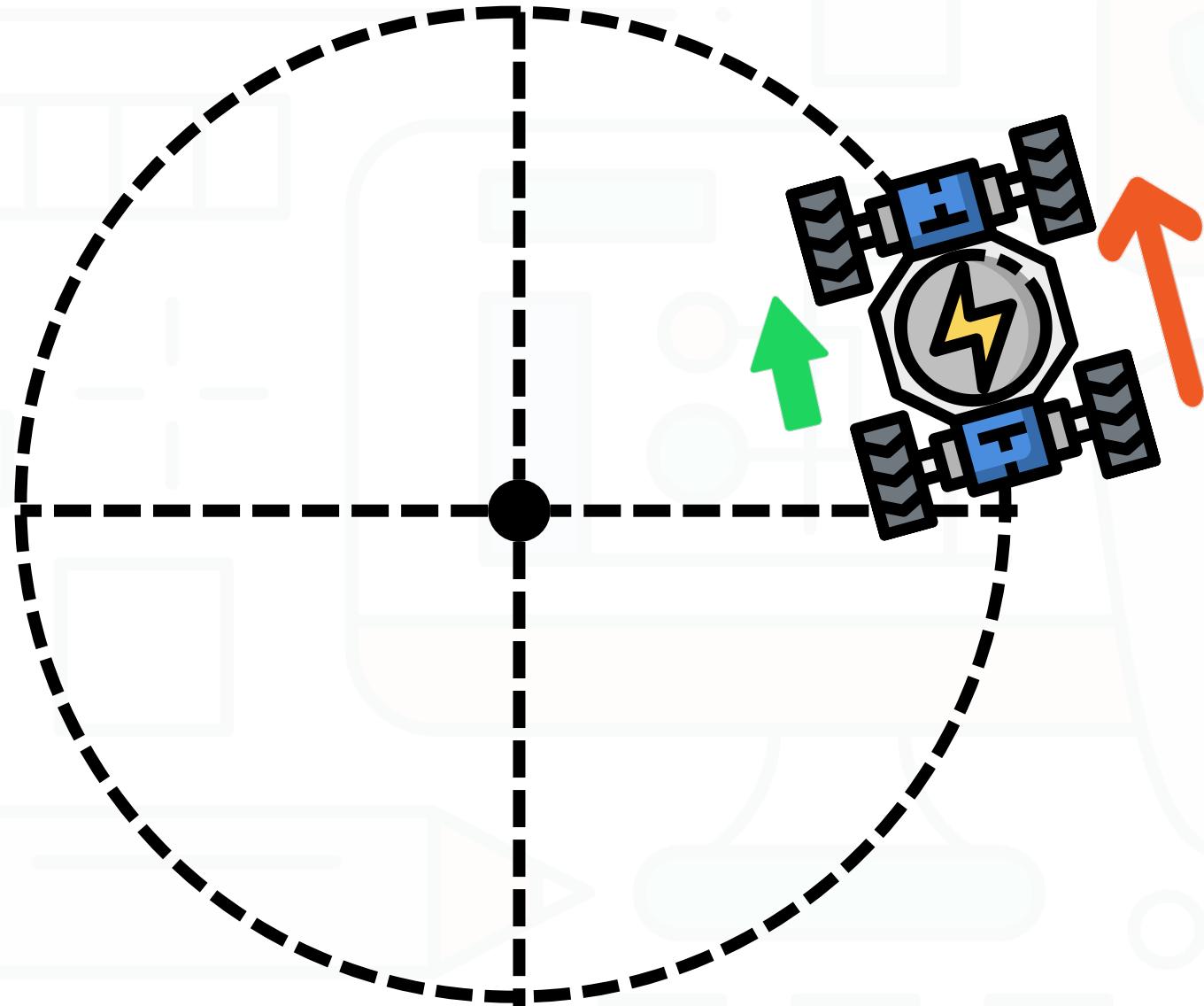
How can we make the robot rotate around an object?



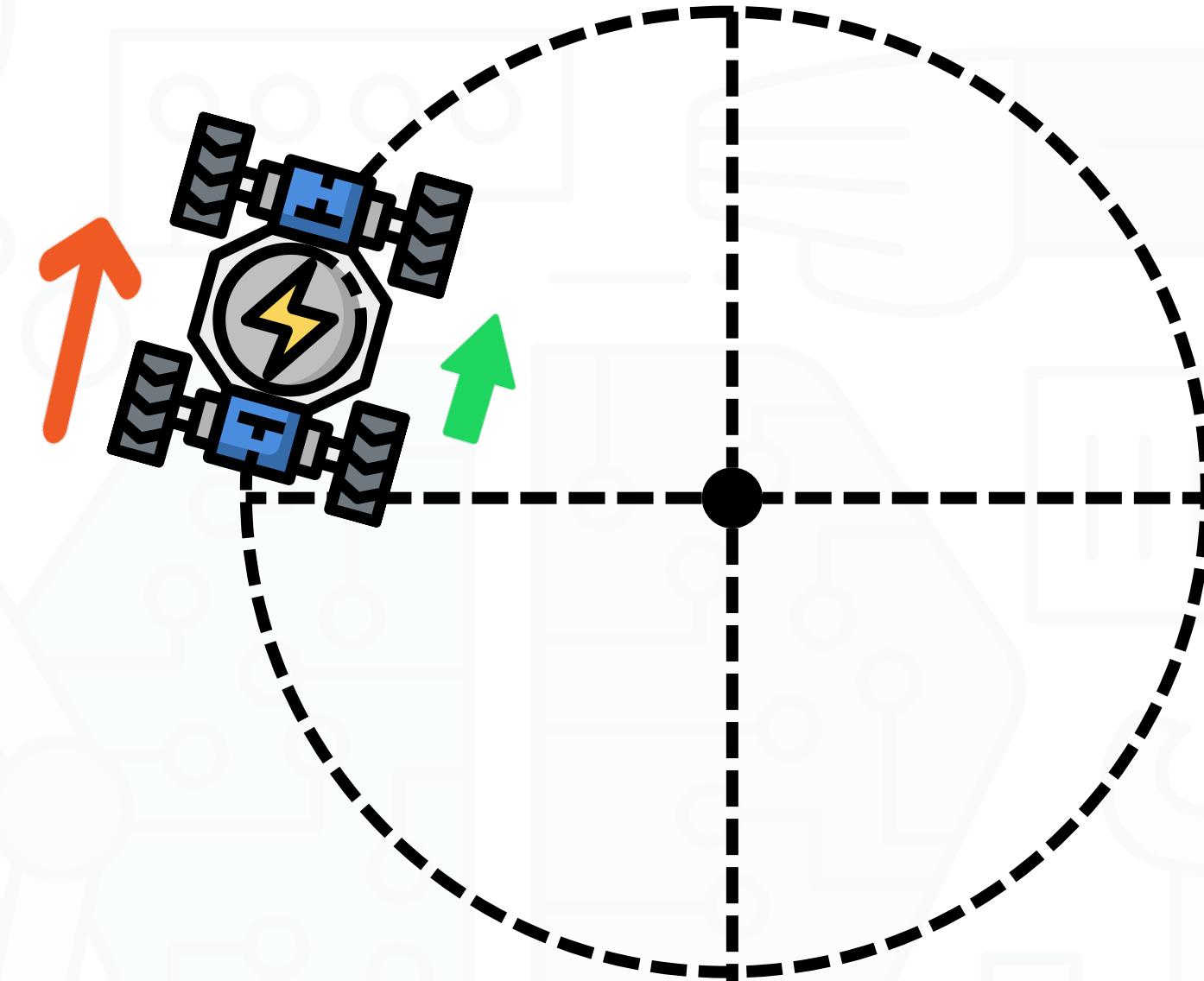
# Rotate The Robot



**Left Rotation**



**Right Rotation**

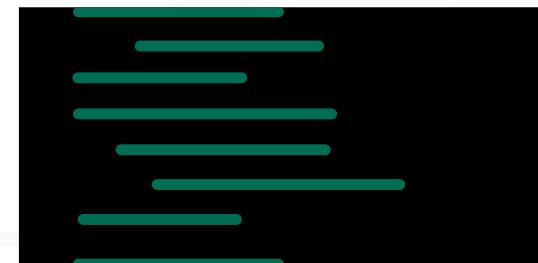


# Let's Code

**Write a code to rotate the robot about its center.**

**Try it by yourself**

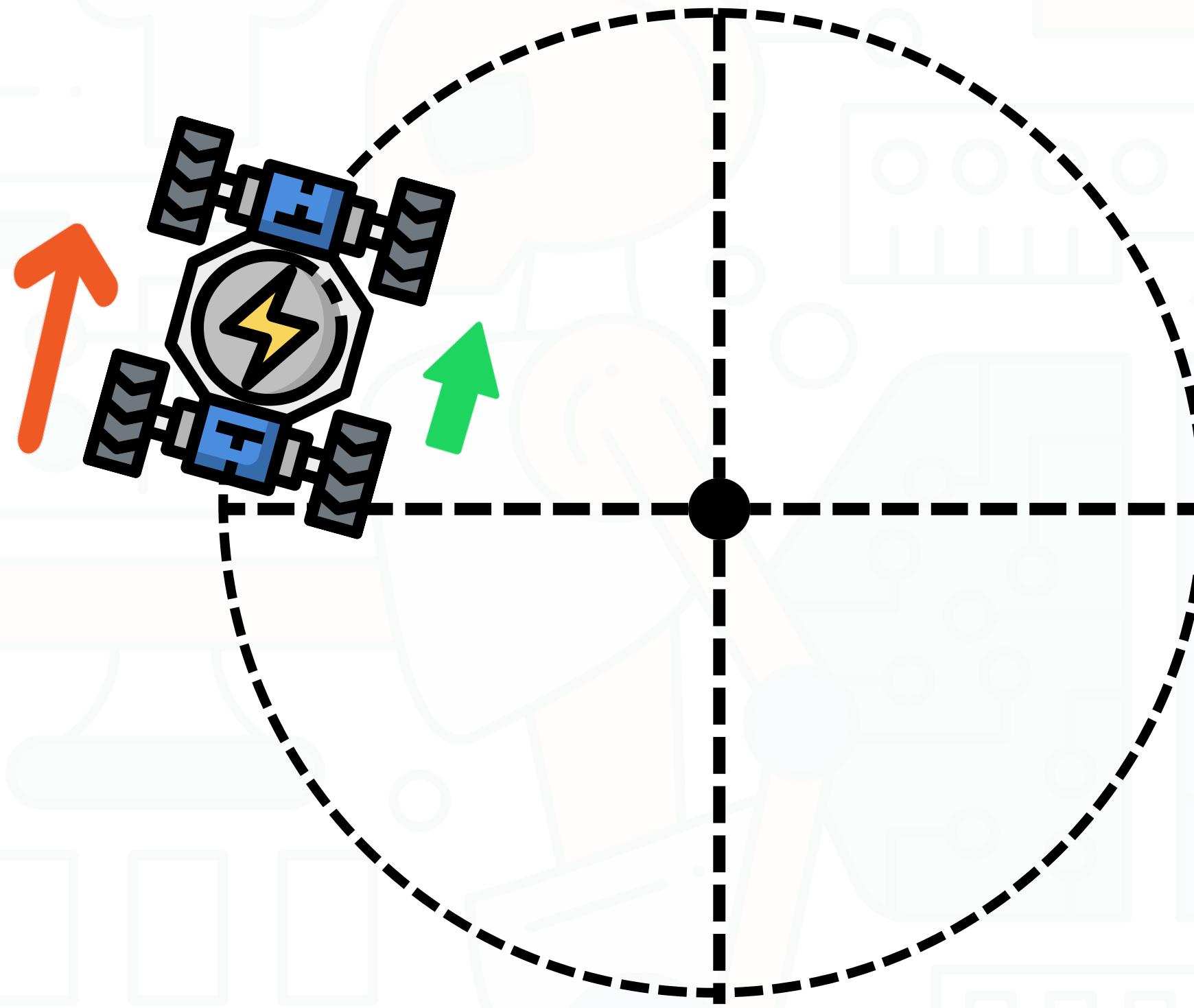
**Hint: Use PWM to set speed.**



# Rotate The Robot



## Right Rotation



# Let's rotate the robot

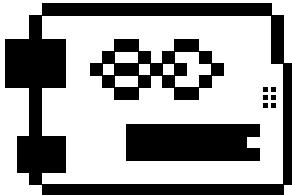


## Step 1: Set motors' speed

Left wheels PWM pins {  
`analogWrite(5, 255);  
analogWrite(6, 255);`

Right wheels PWM pins {  
`analogWrite(9, 125);  
analogWrite(10, 125);`

Notice Left wheels have full speed while right wheel takes half the speed

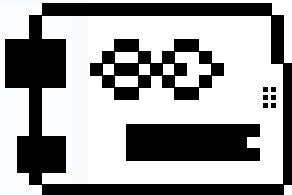


# Let's rotate the robot



## Step 2: Moving the right wheels forward

```
//Back right motor forward  
digitalWrite(13, HIGH);  
digitalWrite(A0, LOW);  
  
//front right motor forward  
digitalWrite(A1, HIGH);  
digitalWrite(A2, LOW);
```

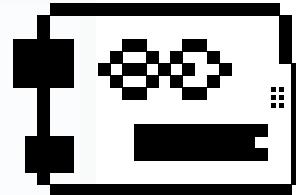


# Let's rotate the robot



## Step 3: Moving the left wheels forward

```
//back left motor forward  
digitalWrite(7, HIGH);  
digitalWrite(8, LOW);  
  
//front left motor forward  
digitalWrite(11, HIGH);  
digitalWrite(12, LOW);
```



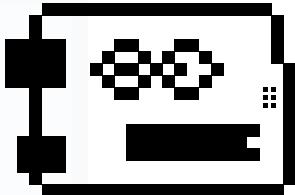
# Let's rotate the robot



## The whole code

```
void setup() {  
    // put your setup code here, to run once:  
  
    //define motors PWM pins  
    pinMode(5, OUTPUT); //back left motor PWM pin  
    pinMode(6, OUTPUT); //front left motor PWM pin  
    pinMode(9, OUTPUT); //back right motor PWM pin  
    pinMode(10, OUTPUT); //front right motor PWM pin  
  
    //setting motors speeds to 255  
  
    analogWrite(5, 255);  
    analogWrite(6, 255);  
  
    analogWrite(9, 125);  
    analogWrite(10, 125);  
  
    //defining motors directions pin  
    pinMode(7, OUTPUT); //back left motor pin 1  
    pinMode(8, OUTPUT); //back left motor pin 2  
  
    pinMode(11, OUTPUT); //front left motor pin 1  
    pinMode(12, OUTPUT); //front left motor pin 2  
  
    pinMode(13, OUTPUT); //back right motor pin 1  
    pinMode(A0, OUTPUT); //back right motor pin 2  
  
    pinMode(A1, OUTPUT); //front right motor pin 1  
    pinMode(A2, OUTPUT); //front right motor pin 2  
}
```

```
void loop() {  
    // put your main code here, to run repeatedly:  
  
    //////////////forward/////////////  
  
    //back left motor forward  
    digitalWrite(7, HIGH);  
    digitalWrite(8, LOW);  
  
    //front left motor forward  
    digitalWrite(11, HIGH);  
    digitalWrite(12, LOW);  
  
    //Back right motor forward  
    digitalWrite(13, HIGH);  
    digitalWrite(A0, LOW);  
  
    //front right motor forward  
    digitalWrite(A1, HIGH);  
    digitalWrite(A2, LOW);  
}
```



# Think

## What are functions?



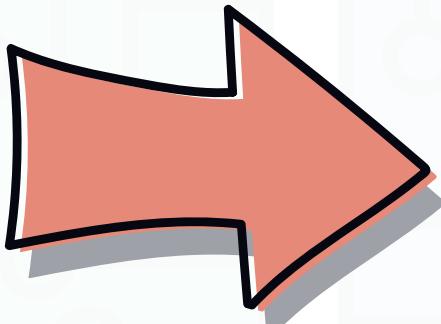
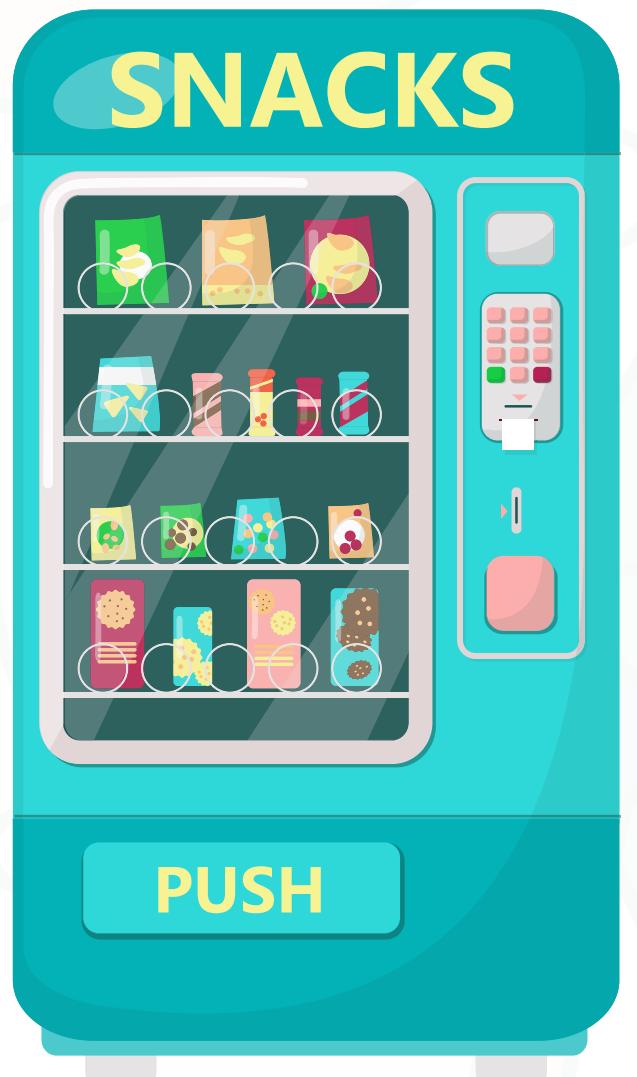
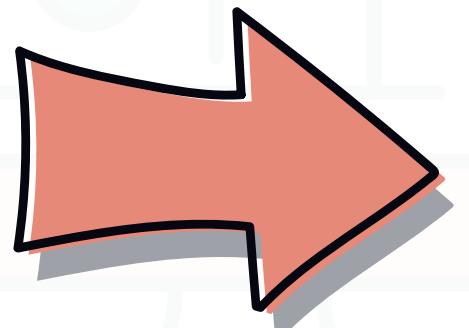
# Functions

What is a function?

A group of actions that takes an input and produces an output



Input



Output

# Analogy

Imagine you have a cake recipe that mixes the ingredients and puts the mix in the oven.

What will be the output if we used eggs, flour, sugar and strawberries as the ingredients?

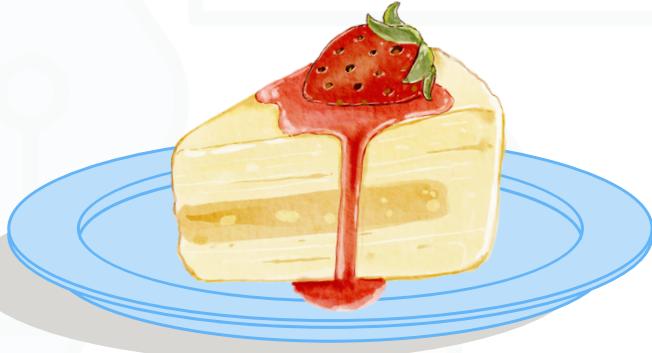
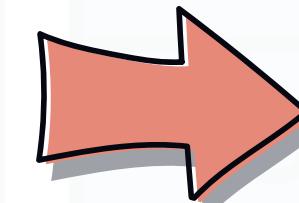
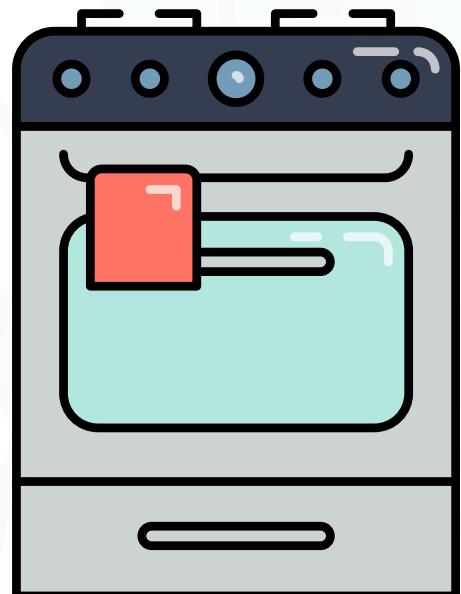


Input

Step 1



Step 2



Output

# Analogy ≡

What will be the output if we used chocolate instead of strawberries?

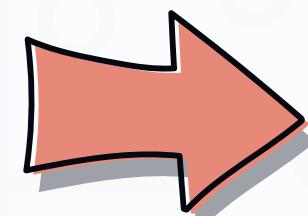
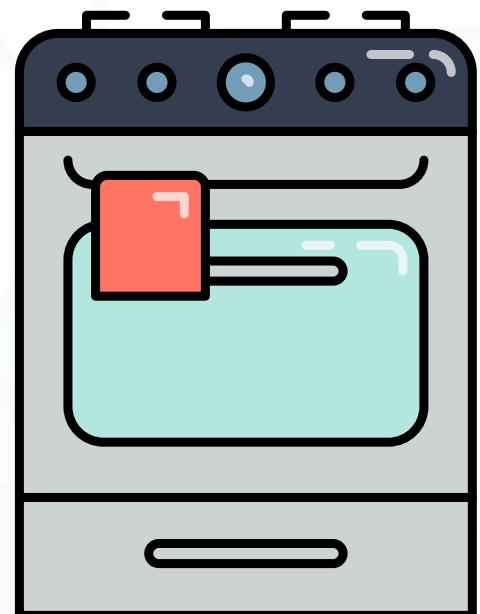


Input

Step 1



Step 2



Output

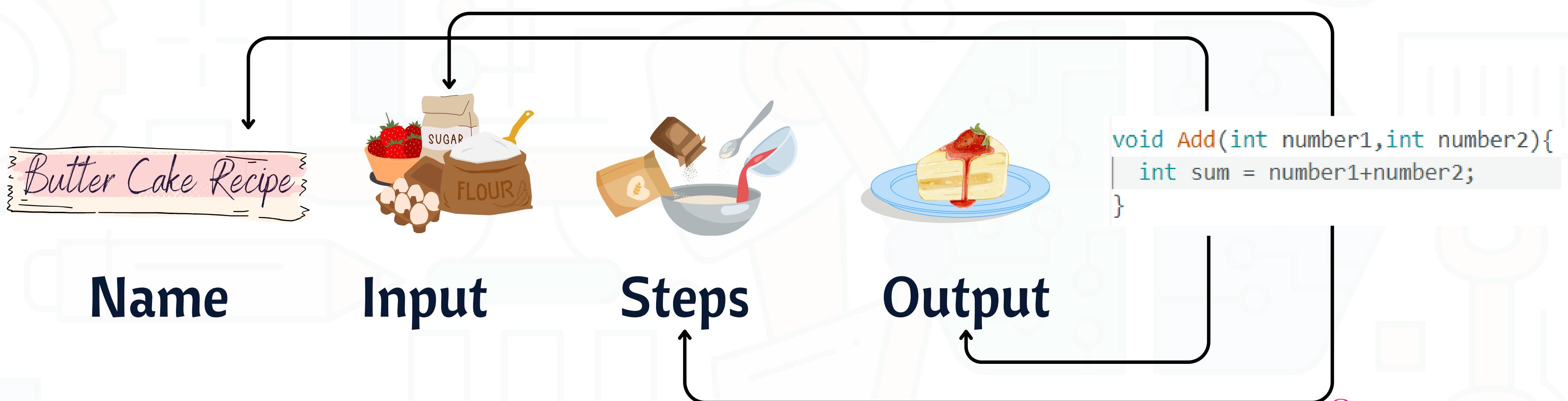
# Think

## How functions help us in coding?



# Analogy ≡

A function in programming is like a recipe for a computer. Just like how a recipe tells you how to make a cake, a function tells the computer what to do. Both have:



# Let's code

**Write a code to move forward and backward**

**Try it by yourself**

**Hint:** Use begin function to set pin modes



# Code



## Define global values names

we can define a name to a value by #define then whenever we use five it will be considered as value 5

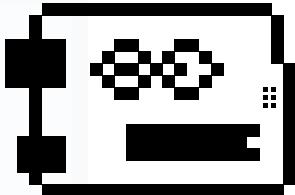
```
#define five 5
```

# Code



## Define motors' pins

```
#define MotorBLS 5 //define back-left-speed-pin number  
#define MotorFLS 6 //define front-left-speed-pin number  
#define MotorBRS 9 //define back-right-speed-pin number  
#define MotorFRS 10 //define front-right-speed-pin number  
#define MotorBL1 7 //define back-left-direction1-pin number  
#define MotorBL2 8 //define back-left-direction2-pin number  
#define MotorFL1 11 //define front-left-direction1-pin number  
#define MotorFL2 12 //define front-left-direction2-pin number  
#define MotorBR1 13 //define back-right-direction1-pin number  
#define MotorBR2 14 //define back-right-direction2-pin number  
#define MotorFR1 15 //define front-right-direction1-pin number  
#define MotorFR2 16 //define front-right-direction2-pin number
```



# Begin Function

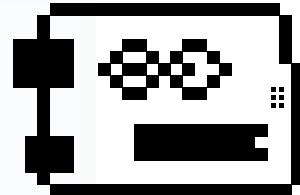
**we define the function  
begin and when called in  
setup, The included codes'  
lines are executed**

# Code



```
void begin (){  
  
    pinMode(MotorBLS, OUTPUT); //back left motor PWM pin  
    pinMode(MotorFLS, OUTPUT); //front left motor PWM pin  
    pinMode(MotorBRS, OUTPUT); //back right motor PWM pin  
    pinMode(MotorFRS, OUTPUT); //front right motor PWM pin  
  
    //defining motors directions pin  
    pinMode(MotorBL1, OUTPUT); //back left motor pin 1  
    pinMode(MotorBL2, OUTPUT); //back left motor pin 2  
  
    pinMode(MotorFL1, OUTPUT); //front left motor pin 1  
    pinMode(MotorFL2, OUTPUT); //front left motor pin 2  
  
    pinMode(MotorBR1, OUTPUT); //back right motor pin 1  
    pinMode(MotorBR2, OUTPUT); //back right motor pin 2  
  
    pinMode(MotorFR1, OUTPUT); //front right motor pin 1  
    pinMode(MotorFR2, OUTPUT); //front right motor pin 2  
}
```

```
void setup() {  
    // put your setup code here, to run once:  
    begin();  
}
```



# Code



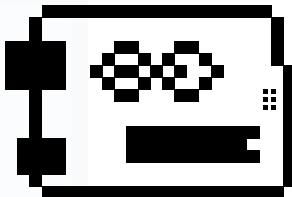
## Forward Function

Forward(175);

**Forward function to move  
the robot forward**

```
void Forward (int speed)
{
    //setting motors speeds to 0/255
    analogWrite(MotorBLS, speed);
    analogWrite(MotorFLS, speed);
    analogWrite(MotorBRS, speed);
    analogWrite(MotorFRS, speed);

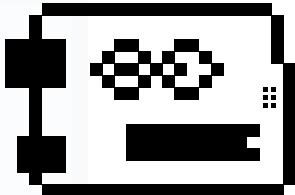
    ///////////////forward/////////////
    //front left motor
    digitalWrite(MotorFL1, HIGH);
    digitalWrite(MotorFL2, LOW);
    //back right motor
    digitalWrite(MotorBR1, HIGH);
    digitalWrite(MotorBR2, LOW);
    //front right motor
    digitalWrite(MotorFR1, HIGH);
    digitalWrite(MotorFR2, LOW);
    //back left motor
    digitalWrite(MotorBL1, HIGH);
    digitalWrite(MotorBL2, LOW);
}
```



# Code



```
void setup() {  
    // put your setup code here, to run once:  
    begin();  
}  
void loop() {  
    // put your main code here, to run repeatedly:  
    Forward(170);  
    delay(5000);  
    Stop();  
    Backward(170);  
    delay(5000);  
    Stop();  
}
```



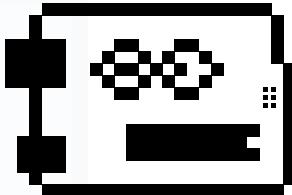
# Let's make functions

## Step 1: Define the function

```
void Test(int number)
```



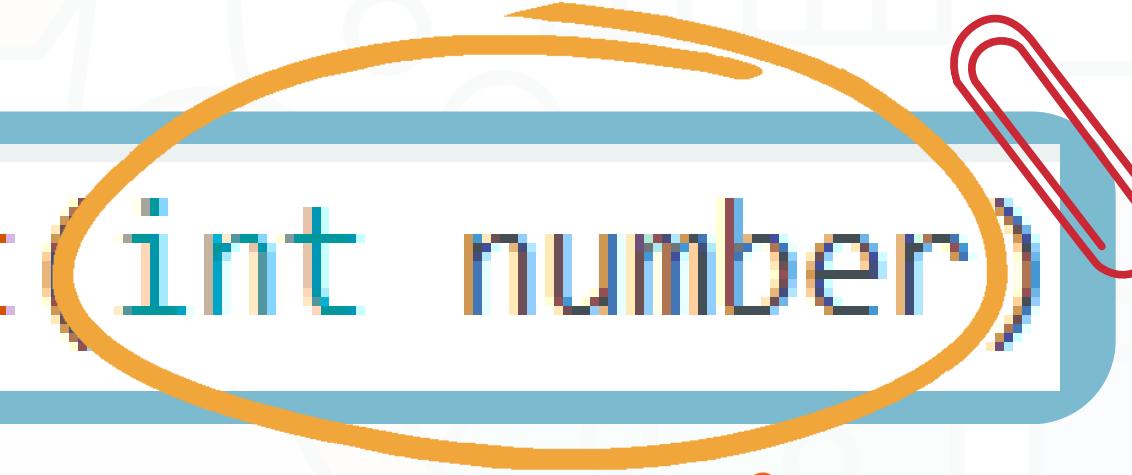
void means the function  
doesn't give us an output



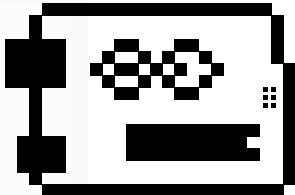
# Let's make functions

## Step 2: Define parameters

```
void Test(int number)
```



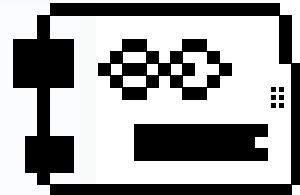
define parameter



# Let's make functions

## Step 3: Build the function

```
void Test(int number){  
    pinMode(number,OUTPUT);  
}
```

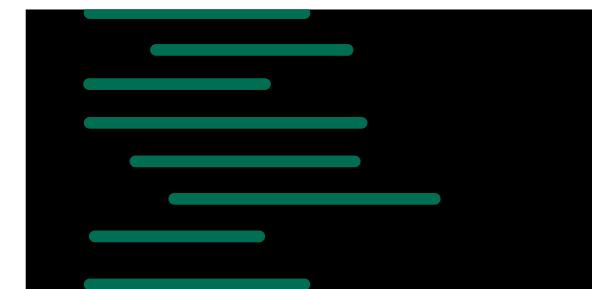


# Code

**Write a code to rotate as a function**

**Try it by yourself**

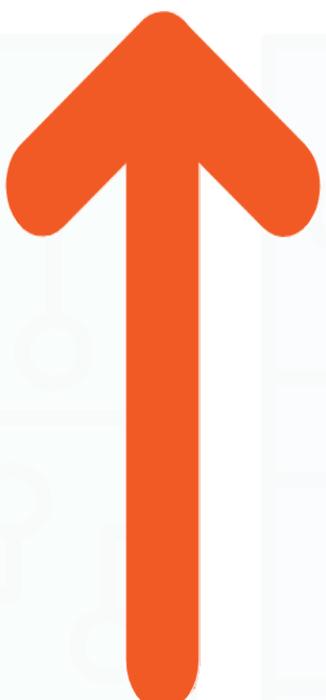
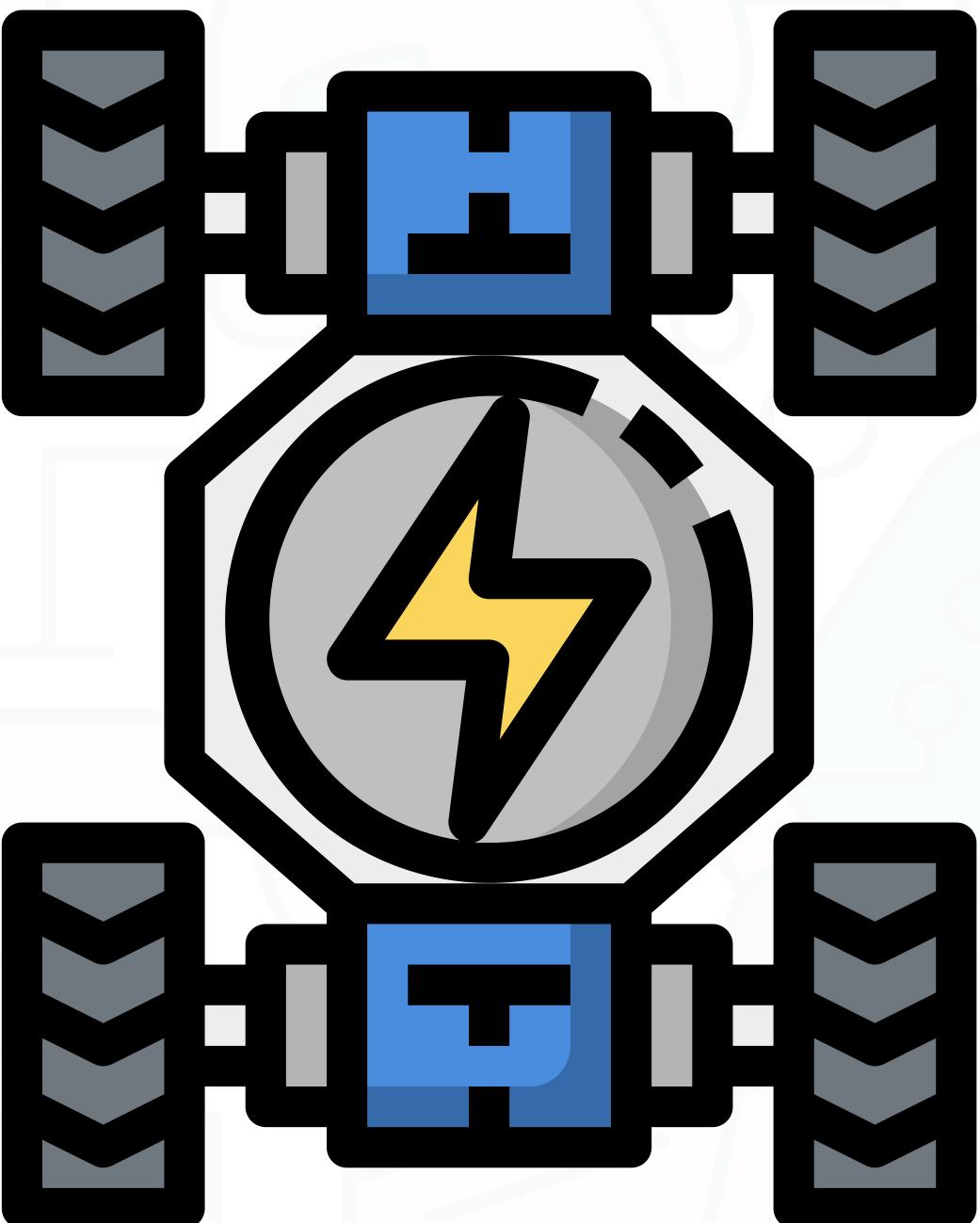
**Hint: Use begin function to Define pins.**



# Rotate The Robot

## Left Rotation

- Enable -> D6
- Input 1 -> D11
- Input2 -> D12



- Enable -> D10
- Input 1 -> A1(D15)
- Input2 -> A2(D16)

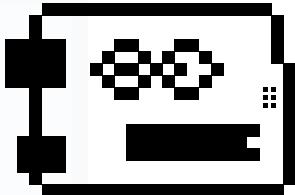
- Enable -> D5
- Input 1 -> D7
- Input2 -> D8

- Enable -> D9
- Input 1 -> D13
- Input2 -> A0(D14)

# Let's make functions

Step 1: Define the function and parameters

```
void rotateLeft (int speed)
```



# Let's make functions

## Step 2: Build the function lines

**Set motors' speed**

**Move the left  
wheels backward**

**Move the right  
wheels forward**

```
void rotateLeft (int speed)
{
    //setting motors speeds to 0/255
    analogWrite(MotorBLS,speed);
    analogWrite(MotorFLS,speed);
    analogWrite(MotorBRS,speed);
    analogWrite(MotorFRS,speed);

    //Back left motor backward
    digitalWrite(MotorBL1,LOW);
    digitalWrite(MotorBL2,HIGH);
    //front left motor backward
    digitalWrite(MotorFL1,LOW);
    digitalWrite(MotorFL2,HIGH);
    //front right motor forward
    digitalWrite(MotorBR1,HIGH);
    digitalWrite(MotorBR2,LOW);
    //front right motor forward
    digitalWrite(MotorFR1,HIGH);
    digitalWrite(MotorFR2,LOW);
}
```