

SPIKE™ Prime LEGO® Education

Programas Python
Soluções possíveis

SPIKE™ Prime LEGO® Education

Socorro!

<https://education.lego.com/pt-br/lessons/prime-invention-squad/help>



```
from spike import PrimeHub, App, ColorSensor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
app = App()
color_sensor = ColorSensor('B')
```

```
# Esta é a história nº 1: Kiki está indo dar uma volta. Ela está passeando feliz quando...
hub.left_button.wait_until_pressed()
```

```
color_sensor.wait_until_color('blue')
app.play_sound('Traffic')
```

```
color_sensor.wait_until_color('yellow')
app.play_sound('Ring Tone')
```

```
color_sensor.wait_until_color('green')
app.play_sound('Dog Bark 1')
app.play_sound('Dog Bark 1')
```

```
# Esta é a história nº 2.
hub.right_button.wait_until_pressed()
```

```
color_sensor.wait_until_color('blue')
app.play_sound('Door Knock')
```

```
color_sensor.wait_until_color('yellow')
app.play_sound('Glass Breaking')
```

```
color_sensor.wait_until_color('green')
app.play_sound('Dog Bark 3')
```

SPIKE™ Prime LEGO® Education

Corrida de Pulga

<https://education.lego.com/pt-br/lessons/prime-invention-squad/hopper-race>



```
from spike import PrimeHub, MotorPair
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
hop_motors = MotorPair('E', 'F')
```

```
hop_motors.set_default_speed(50)
```

```
hub.light_matrix.write('3')
wait_for_seconds(1)
```

```
hub.light_matrix.write('2')
wait_for_seconds(1)
```

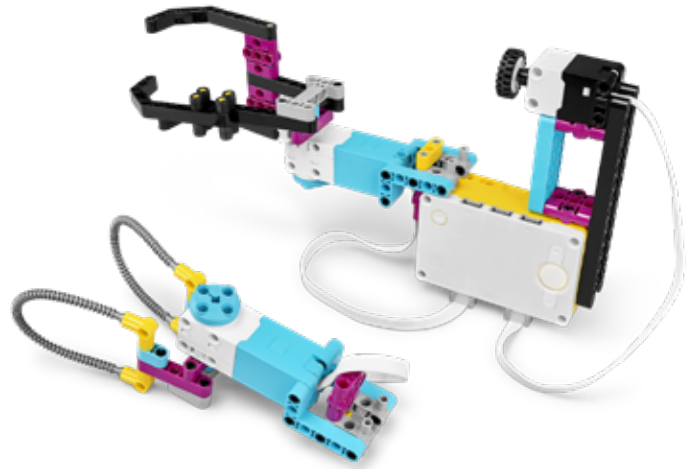
```
hub.light_matrix.write('1')
wait_for_seconds(1)
```

```
# Ajuste isso para alterar a distância que sua pulga irá se mover.
# -----v
hop_motors.move(10, 'seconds')
```

SPIKE™ Prime LEGO® Education

Super Limpeza

<https://education.lego.com/pt-br/lessons/prime-invention-squad/super-cleanup>



```
from spike import ForceSensor, Motor
```

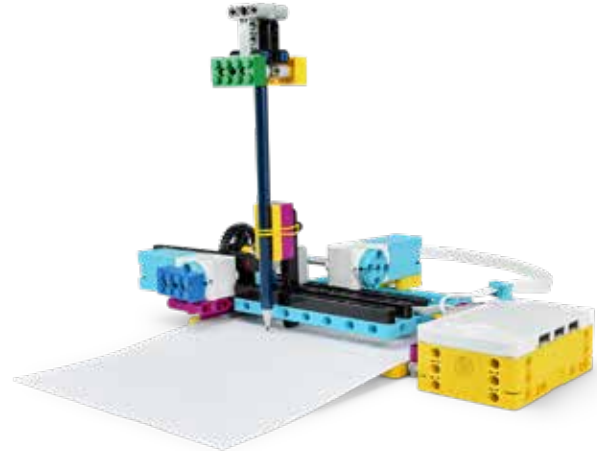
```
force_sensor = ForceSensor('E')  
grabber_motor = Motor('A')
```

```
while True:  
    force_sensor.wait_until_pressed()  
    grabber_motor.set_stall_detection(False)  
    grabber_motor.start(-75)  
  
    force_sensor.wait_until_released()  
    grabber_motor.set_stall_detection(True)  
    grabber_motor.start(75)
```

SPIKE™ Prime LEGO® Education

Quebrado

<https://education.lego.com/pt-br/lessons/prime-invention-squad/broken>



```
from spike import PrimeHub, Motor
from spike.control import wait_for_seconds
hub = PrimeHub()
x_motor = Motor('A')
y_motor = Motor('C')
```

```
hub.left_button.wait_until_pressed()
x_motor.set_default_speed(-100)
x_motor.run_for_seconds(1.5)
wait_for_seconds(1)
```

```
# Estas linhas devem "cortar" um quadrado.
x_motor.set_default_speed(100)
y_motor.set_default_speed(100)
x_motor.run_for_degrees(400)
y_motor.run_for_degrees(575)
x_motor.run_for_degrees(-400)
y_motor.run_for_degrees(-575)
```

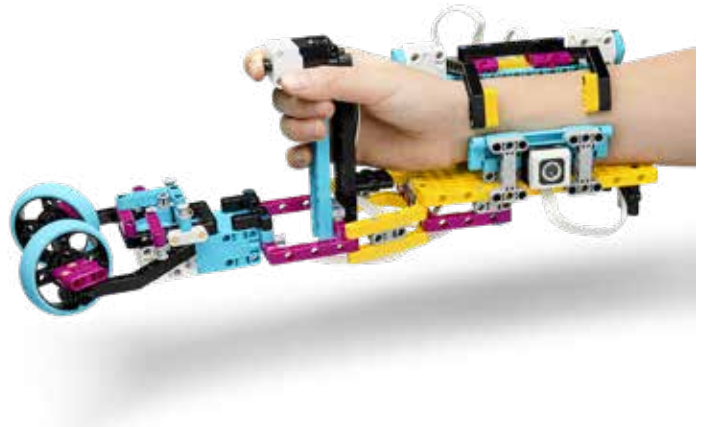
```
hub.right_button.wait_until_pressed()
x_motor.set_default_speed(100)
x_motor.run_for_seconds(1.5)
```

```
wait_for_seconds(1)
# Estas linhas devem "cortar" um retângulo.
x_motor.run_for_degrees(-60)
x_motor.run_for_degrees(-400)
y_motor.run_for_degrees(-800)
x_motor.run_for_degrees(400)
y_motor.run_for_degrees(800)
```

SPIKE™ Prime LEGO® Education

Projetar para Alguém

<https://education.lego.com/pt-br/lessons/prime-invention-squad/design-for-someone>



```
from spike import PrimeHub, Motor, ForceSensor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
motor_a = Motor('A')
motor_e = Motor('E')
force_sensor = ForceSensor('B')
```

```
motor_a.set_default_speed(100)
motor_e.set_default_speed(-100)
motor_a.set_stall_detection(False)
motor_e.set_stall_detection(False)
motor_a.set_stop_action('hold')
motor_e.set_stop_action('hold')
```

```
motor_a.run_to_position(0)
hub.speaker.beep(60)
hub.speaker.beep(72)
```

```
# faça a prótese agarrar o braço de alguém
motor_a.run_for_seconds(1)
motor_e.run_for_seconds(1)
```

```
while True:
    if hub.right_button.was_pressed():
        # faça a prótese soltar
        motor_a.run_to_position(0)
        motor_e.run_to_position(0)
        break

    if force_sensor.get_force_newton() > 5:
        hub.light_matrix.show_image('SQUARE')
    else:
        hub.light_matrix.off()

    wait_for_seconds(0.01)
```

SPIKE™ Prime LEGO® Education

Faça seu Pedido

<https://education.lego.com/pt-br/lessons/prime-kickstart-a-business/place-your-order>



```
from spike import PrimeHub, App, ColorSensor, DistanceSensor, Motor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
app = App()
distance_sensor = DistanceSensor('C')
color_sensor = ColorSensor('D')
arm_motor = Motor('A')
base_motor = Motor('F')
```

```
arm_motor.set_default_speed(50)
base_motor.set_default_speed(50)
```

```
arm_motor.run_to_position(350)
base_motor.run_to_position(350)
```

```
app.start_sound('Connect')
distance_sensor.light_up_all()
```

```
for x in range(10):
    hub.light_matrix.show_image('HEART')
    wait_for_seconds(0.5)
    hub.light_matrix.show_image('HEART_SMALL')
    wait_for_seconds(0.5)
```

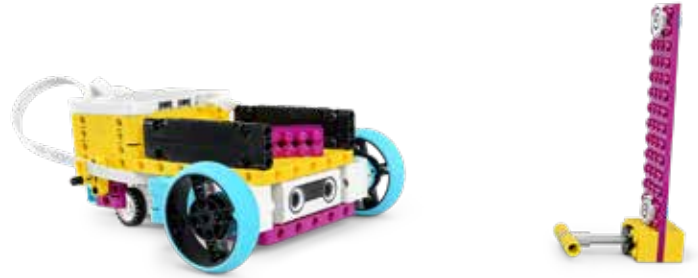
```
hub.light_matrix.show_image('HEART')
```

```
while True:
    color_sensor.wait_until_color('violet')
    arm_motor.run_for_degrees(30)
    arm_motor.run_for_degrees(-60)
    arm_motor.run_for_degrees(60)
    arm_motor.run_for_degrees(-30)
    app.start_sound('Connect')
    hub.light_matrix.show_image('HEART')
```

SPIKE™ Prime LEGO® Education

Fora de Serviço

<https://education.lego.com/pt-br/lessons/prime-kickstart-a-business/out-of-order>



```

from spike import PrimeHub, DistanceSensor, Motor, MotorPair
from spike.control import wait_for_seconds

hub = PrimeHub()
distance_sensor = DistanceSensor('B')
drive_motors = MotorPair('A', 'E')
small_wheel_motor = Motor('C')

small_wheel_motor.set_default_speed(100)
drive_motors.set_default_speed(50)

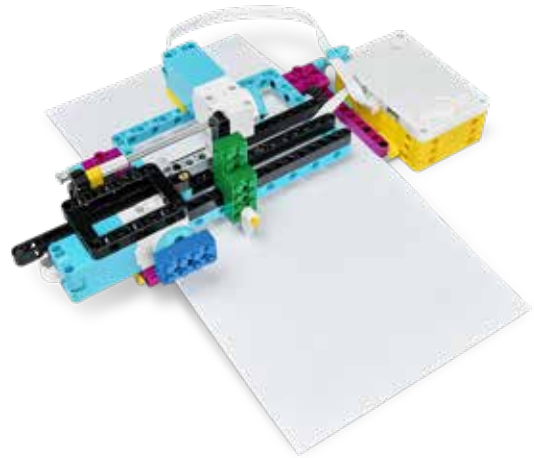
hub.left_button.wait_until_pressed()
# Esta é uma maneira de depurar o primeiro programa.
small_wheel_motor.run_to_position(0)
drive_motors.start()
# ajuste o valor aqui -----v
distance_sensor.wait_for_distance_closer_than(15, DistanceSensor.CM)
drive_motors.stop()

hub.right_button.wait_until_pressed()
# Esta é uma maneira de depurar o segundo programa.
small_wheel_motor.run_to_position(0)
drive_motors.start()
# ajuste o valor aqui -----v
distance_sensor.wait_for_distance_closer_than(15, DistanceSensor.CM)
drive_motors.stop()
# ajuste o valor aqui -----v
small_wheel_motor.run_to_position(20)
wait_for_seconds(1)
drive_motors.move(-50, DistanceSensor.CM)
drive_motors.stop()
small_wheel_motor.run_to_position(0)
wait_for_seconds(1)
# ajuste o valor aqui
# -----v
drive_motors.move(50, DistanceSensor.CM)
drive_motors.stop()
  
```


SPIKE™ Prime LEGO® Education

Acompanhe seus Pacotes

<https://education.lego.com/pt-br/lessons/prime-kickstart-a-business/track-your-packages>



```
from spike import PrimeHub, Motor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
horizontal_motor = Motor('A')
vertical_motor = Motor('C')
```

```
horizontal_motor.set_default_speed(75)
vertical_motor.set_default_speed(75)
```

```
# Este programa irá rastrear seu pacote no mapa nº 1.
```

```
hub.left_button.wait_until_pressed()
horizontal_motor.run_for_seconds(1)
wait_for_seconds(1)
```

```
vertical_motor.run_for_degrees(475)
horizontal_motor.run_for_degrees(-545)
vertical_motor.run_for_degrees(950)
horizontal_motor.run_for_degrees(550)
vertical_motor.run_for_degrees(380)
```

```
# execute os dois motores ao mesmo tempo para mover-se diagonalmente
```

```
vertical_motor.start(speed=75)
horizontal_motor.run_for_degrees(-540, speed=50)
vertical_motor.stop()
```

```
vertical_motor.run_for_degrees(175)
```

SPIKE™ Prime LEGO® Education

Mantenha seguro

<https://education.lego.com/pt-br/lessons/prime-kickstart-a-business/keep-it-safe>



```
from spike import PrimeHub, Motor, LightMatrix
from spike.control import wait_for_seconds, wait_until
from spike.operator import greater_than
```

```
hub = PrimeHub()
lock_motor = Motor('C')
dial_motor = Motor('B')
lock_motor.set_default_speed(50)
```

```
hub.speaker.beep(60)
hub.speaker.beep(72)
```

```
# Isso tranca a porta.
dial_motor.set_stop_action('coast')
dial_motor.run_to_position(0)
dial_motor.set_degrees_counted(0)
hub.light_matrix.show_image('NO')
```

```
# Isso destranca a porta quando o botão esquerdo do Hub é pressionado.
```

```
hub.left_button.wait_until_pressed()
hub.speaker.beep(72)
wait_until(dial_motor.get_degrees_counted, greater_than, 180)
hub.speaker.beep(60)
lock_motor.run_for_seconds(1)
hub.light_matrix.show_image('NO')
wait_for_seconds(2)
hub.light_matrix.show_image('YES')
wait_for_seconds(5)
```

SPIKE™ Prime LEGO® Education

Mantenha realmente seguro!

<https://education.lego.com/pt-br/lessons/prime-kickstart-a-business/keep-it-really-safe>



```

from spike import PrimeHub, App, Motor
from spike.control import Timer, wait_for_seconds

hub = PrimeHub()
app = App()
dial = Motor('B')
lock = Motor('C')
dial_cover = Motor('E')
timer = Timer()

dial.set_default_speed(75)
lock.set_default_speed(75)
dial_cover.set_default_speed(75)

def unlock():

while not hub.left_button.is_pressed() and dial.get_degrees_counted() < 180:
    hub.speaker.beep(60)
    dial_cover.run_for_degrees(15)
    wait_for_seconds(0.8)

    if timer.now() > 5:
        app.play_sound('Bonk')
        return

    hub.light_matrix.show_image('NO')
    wait_for_seconds(2)
    hub.light_matrix.show_image('YES')
    dial_cover.run_to_position(0)
    lock.run_for_seconds(1)
    app.play_sound('Wand')
    wait_for_seconds(5)

# Isso tranca a porta e inicia o mecanismo de proteção extra.

hub.speaker.beep(60)
hub.speaker.beep(72)
lock.run_for_seconds(-1)
dial.run_to_position(0)
dial_cover.run_to_position(0)
dial.set_degrees_counted(0)
dial.set_stop_action('coast')
hub.light_matrix.show_image('NO')
timer.reset()
unlock()

```

SPIKE™ Prime LEGO® Education

Automatize!

<https://education.lego.com/pt-br/lessons/prime-kickstart-a-business/automate-it>



```

from spike import App, Motor, ColorSensor
from spike.control import wait_for_seconds

app = App()
base_motor = Motor('A')
arm_motor = Motor('F')
color_sensor = ColorSensor('D')

base_motor.set_default_speed(25)
arm_motor.set_default_speed(25)

def check_color():
    # Isso irá verificar a cor do pacote.
    arm_motor.run_to_position(235)
    wait_for_seconds(4)
    if color_sensor.get_color() == 'violet':
        base_motor.run_to_position(0)
        arm_motor.run_to_position(25)
        app.play_sound('Triumph')
        arm_motor.run_to_position(240)
    else:
        app.play_sound('Oops')
        arm_motor.run_to_position(25)
        for x in range(3):
            arm_motor.run_for_degrees(-100, speed=100)
            arm_motor.run_for_degrees(100, speed=100)

# Isso liga o robô e faz ele agarrar um pacote de cada um dos lados.
base_motor.run_to_position(0)
arm_motor.run_to_position(240)

base_motor.run_to_position(90)
arm_motor.run_to_position(25)

check_color()

base_motor.run_to_position(0)
arm_motor.run_to_position(240)
base_motor.run_to_position(270)
arm_motor.run_to_position(25)

check_color()

base_motor.run_to_position(0)
arm_motor.run_to_position(240)

```

SPIKE™ Prime LEGO® Education

Break Dance

<https://education.lego.com/pt-br/lessons/prime-life-hacks/break-dance>



A aula "Break Dance" foi criada para nossa linguagem de programação de blocos de palavras. Atualmente, não é possível usar nossa linguagem de programação Python e seguir o mesmo fluxo da lição.

Porém, aqui está um programa que fará seu modelo se mover!

```
from spike import PrimeHub, Motor, ColorSensor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
leg_motor = Motor('F')
arm_motor = Motor('B')
color_sensor = ColorSensor('D')
```

```
leg_motor.set_default_speed(-80)
arm_motor.set_default_speed(-80)
```

```
leg_motor.run_to_position(0)
arm_motor.run_to_position(0)
wait_for_seconds(1)
```

```
for x in range(10):
    hub.light_matrix.write("1")
    leg_motor.start()
    arm_motor.run_for_rotations(1)
    leg_motor.stop()
    wait_for_seconds(0.45)
```

```
    hub.light_matrix.write("2")
    leg_motor.start()
    arm_motor.run_for_rotations(1)
    leg_motor.stop()
    wait_for_seconds(0.45)
```

```
    hub.light_matrix.write("3")
    leg_motor.start()
    arm_motor.run_for_rotations(1)
    leg_motor.stop()
    wait_for_seconds(0.45)
```

SPIKE™ Prime LEGO® Education

Repetir 5 Vezes

<https://education.lego.com/pt-br/lessons/prime-life-hacks/repeat-5-times>



```

from spike import PrimeHub, App, Motor
from spike.control import wait_until, wait_for_seconds
from spike.operator import equal_to

hub = PrimeHub()
app = App()
left_leg_motor = Motor('B')
right_leg_motor = Motor('F')
left_leg_motor.set_default_speed(50)
right_leg_motor.set_default_speed(-50)
left_leg_motor.start()
right_leg_motor.start()

wait_until(hub.motion_sensor.get_orientation, equal_to, 'leftside')
right_leg_motor.stop()
left_leg_motor.stop()

app.play_sound('Sport Whistle 1')

for count in range(5):
    left_leg_motor.set_default_speed(-50)
    right_leg_motor.set_default_speed(50)
    left_leg_motor.start()
    right_leg_motor.start()
    wait_until(hub.motion_sensor.get_orientation, equal_to, 'front')
    right_leg_motor.stop()
    left_leg_motor.stop()
    app.start_sound('Male Jump 1')
    hub.light_matrix.write(count + 1)
    wait_for_seconds(0.5)
    left_leg_motor.set_default_speed(50)
    right_leg_motor.set_default_speed(-50)
    left_leg_motor.start()
    right_leg_motor.start()
    wait_until(hub.motion_sensor.get_orientation, equal_to, 'leftside')
    right_leg_motor.stop()
    left_leg_motor.stop()
    wait_for_seconds(0.5)

app.play_sound('Sport Whistle 2')

```

SPIKE™ Prime LEGO® Education

Chuva ou sol?

<https://education.lego.com/pt-br/lessons/prime-life-hacks/rain-or-shine>



A aula "Chuva ou sol?" foi criada para nossa linguagem de programação de blocos de palavras. Atualmente, não é possível usar as funções de previsão do tempo com nossa linguagem de programação Python.

Porém, aqui está um programa que fará com que o meteorologista se mova!

```
from spike import PrimeHub, App, Motor
from spike.control import wait_for_seconds

hub = PrimeHub()
app = App()
umbrella_motor = Motor("B")
glasses_motor = Motor("F")
YOUR_LOCAL_FORECAST = "sunny"

umbrella_motor.set_default_speed(100)
glasses_motor.set_default_speed(100)

# Isso coloca o robô na posição inicial correta.
umbrella_motor.run_to_position(45)
glasses_motor.run_to_position(300)

hub.speaker.beep(60, seconds=0.1)
hub.speaker.beep(72, seconds=0.1)

if YOUR_LOCAL_FORECAST == "sunny":
    # se estiver ensolarado, coloque óculos escuros
    glasses_motor.run_to_position(0)
    hub.light_matrix.show_image("SQUARE")
    wait_for_seconds(2)
    glasses_motor.run_to_position(300)
elif YOUR_LOCAL_FORECAST == "rainy":
    # ou se estiver chovendo, abra o guarda-chuva
    umbrella_motor.run_to_position(340)
    app.play_sound("Rain")
    umbrella_motor.run_to_position(45)
else:
    # caso contrário, mostrar "X"
    hub.light_matrix.show_image("N0")
```

SPIKE™ Prime LEGO® Education

Velocidade do Vento

<https://education.lego.com/pt-br/lessons/prime-life-hacks/wind-speed>



A aula "Velocidade do Vento" foi criada para nossa linguagem de programação de blocos de palavras. Atualmente, não é possível usar as funções de previsão do tempo com nossa linguagem de programação Python.

Porém, aqui está um programa que fará seu modelo se mover!

```
from spike import App, Motor
from spike.control import wait_for_seconds
```

```
tilt_motor = Motor("A")
WIND_SPEED_FORECAST = 8
```

```
tilt_motor.set_default_speed(20)
tilt_motor.run_to_position(5)
```

```
if WIND_SPEED_FORECAST < 5.5:
    tilt_motor.run_for_degrees(30)
    wait_for_seconds(1)
    tilt_motor.run_for_degrees(-30)
else:
    tilt_motor.run_for_degrees(60)
    wait_for_seconds(1)
    tilt_motor.run_for_degrees(-60)
```


SPIKE™ Prime LEGO® Education

Amor Vegano

<https://education.lego.com/pt-br/lessons/prime-life-hacks/veggie-love>



A aula " Amor Vegano " foi criada para nossa linguagem de programação de blocos de palavras. Atualmente, não é possível usar as funções de previsão do tempo com nossa linguagem de programação Python.

Porém, aqui está um programa que fará seu modelo se mover!

```

from spike import PrimeHub, App, Motor

hub = PrimeHub()
app = App()
pointer_motor = Motor("E")
pointer_motor.set_default_speed(-50)

WEEK_RAIN = 50
ROTATION = 0

hub.left_button.wait_until_pressed()
pointer_motor.run_for_seconds(2)
pointer_motor.set_degrees_counted(0)
pointer_motor.set_default_speed(50)
pointer_motor.run_for_seconds(2)
hub.light_matrix.write(abs(pointer_motor.get_degrees_counted()))
rotation = int(week_rain * abs(pointer_motor.get_degrees_counted()) / 60)
print(ROTATION)

hub.right_button.wait_until_pressed()
pointer_motor.set_degrees_counted(0)
pointer_motor.set_default_speed(-50)
pointer_motor.run_for_degrees(ROTATION)
hub.light_matrix.write(WEEK_RAIN)
print(WEEK_RAIN)

```

SPIKE™ Prime LEGO® Education

Jogo do Cérebro

<https://education.lego.com/pt-br/lessons/prime-life-hacks/brain-game>



```
from spike import PrimeHub, App, Motor, ColorSensor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
app = App()
mouth_motor = Motor('A')
color_sensor = ColorSensor('B')
candy1 = []
candy2 = []
```

```
while True:
    hub.left_button.wait_until_pressed()

    # Isso irá fazer o Mestre do Jogo comer o bastão
    de doces e depois ler e registrar sua sequência de
    cores na lista chamada "Doces1".
    hub.light_matrix.off()
    candy1.clear()
    mouth_motor.set_default_speed(-50)
    mouth_motor.run_for_seconds(2)
    app.play_sound('Bite')
    app.play_sound('Bite')
```

```
for x in range(5):
    candy1.append(color_sensor.get_color())
    wait_for_seconds(1)
    mouth_motor.set_default_speed(50)
    mouth_motor.run_for_degrees(95)
    wait_for_seconds(1)
```

```
hub.right_button.wait_until_pressed()
```

```
# Isso irá fazer o Mestre do Jogo comer o bastão de
doces e depois ler e registrar sua sequência de cores
na lista chamada "Doces2".
```

```
candy2.clear()
mouth_motor.set_default_speed(-50)
mouth_motor.run_for_seconds(2)
app.play_sound('Bite')
app.play_sound('Bite')
```

```
for x in range(5):
    candy2.append(color_sensor.get_color())
```

```
wait_for_seconds(1)
mouth_motor.set_default_speed(50)
mouth_motor.run_for_degrees(95)
wait_for_seconds(1)
```

```
# Ilumine a posição das peças vermelhas se estiverem
na mesma posição em ambos os bastões de doces.
```

```
candy1_red_index = candy1.index('red')
candy2_red_index = candy2.index('red')
for x in range(5):
    print(candy1[x])
```

```
if candy1_red_index == candy2_red_index:
    for x in range(5):
        hub.light_matrix.set_pixel(x, candy1_red_
index)
        app.play_sound('Win')
else:
    app.play_sound('Oops')
```

SPIKE™ Prime LEGO® Education

O Técnico

<https://education.lego.com/pt-br/lessons/prime-life-hacks/the-coach>



```
from spike import Motor
from spike.control import Timer, wait_for_seconds

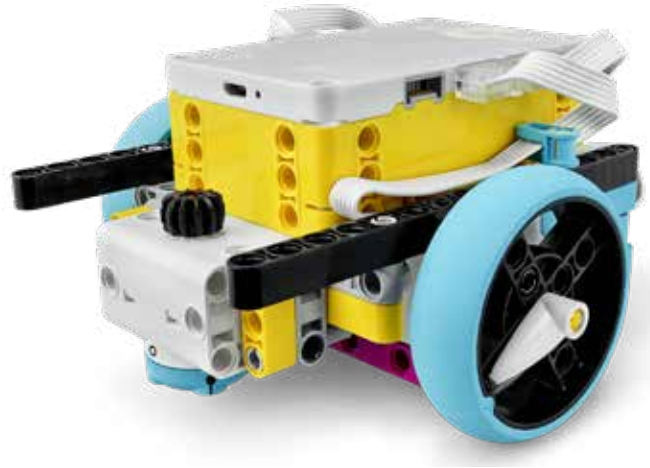
left_leg_motor = Motor('F')
right_leg_motor = Motor('B')
timer = Timer()
left_leg_motor.run_to_position(0)
right_leg_motor.run_to_position(0)

while True:
    while timer.now() < 5:
        left_leg_motor.start_at_power(-80)
        right_leg_motor.start_at_power(80)
        wait_for_seconds(0.1)
        left_leg_motor.start_at_power(80)
        right_leg_motor.start_at_power(-80)
        wait_for_seconds(0.1)
```

SPIKE™ Prime LEGO® Education

Acampamento de Treinamento 1

<https://education.lego.com/pt-br/lessons/prime-competition-ready/training-camp-1-driving-around>



```
from spike import MotorPair
from spike.control import wait_for_seconds

drive_motors = MotorPair('C', 'D')

drive_motors.set_default_speed(30)
drive_motors.set_motor_rotation(17.5, 'cm')

wait_for_seconds(1)

for x in range(4):
    drive_motors.move(10, 'cm')
    wait_for_seconds(0.5)
    drive_motors.move(182, 'degrees', steering=100)
```

SPIKE™ Prime LEGO® Education

Acampamento de Treinamento 2

<https://education.lego.com/pt-br/lessons/prime-competition-ready/training-camp-2-playing-with-objects>



```
from spike import PrimeHub, MotorPair, Motor, DistanceSensor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
drive_motors = MotorPair('C', 'D')
grabber_motor = Motor('E')
distance_sensor = DistanceSensor('F')
```

```
drive_motors.set_default_speed(30)
drive_motors.set_motor_rotation(17.5, 'cm')
grabber_motor.set_default_speed(-20)
grabber_motor.run_for_seconds(1)
grabber_motor.set_default_speed(20)
grabber_motor.run_for_degrees(75)
```

```
hub.speaker.beep(60)
hub.speaker.beep(72)
```

```
hub.right_button.wait_until_pressed()
```

```
wait_for_seconds(1)
```

```
drive_motors.start()
distance_sensor.wait_for_distance_closer_than(10, 'cm')
drive_motors.stop()
```

```
grabber_motor.run_for_degrees(-75)
```

```
hub.speaker.beep(60)
hub.speaker.beep(72)
```

```
drive_motors.move(-20, 'cm')
```

SPIKE™ Prime LEGO® Education

Acampamento de Treinamento 3

<https://education.lego.com/pt-br/lessons/prime-competition-ready/training-camp-3-react-to-lines>



```
from spike import PrimeHub, MotorPair, ColorSensor
from spike.control import wait_for_seconds

hub = PrimeHub()
drive_motors = MotorPair('C', 'D')
color_sensor = ColorSensor('B')

drive_motors.set_default_speed(50)
POWER = 50

while True:
    if hub.left_button.was_pressed():
        drive_motors.start()
        color_sensor.wait_until_color('black')
        drive_motors.stop()

    if hub.right_button.was_pressed():
        while True:
            drive_motors.start_tank_at_power(0, POWER)
            color_sensor.wait_until_color('black')
            drive_motors.start_tank_at_power(POWER, 0)
            color_sensor.wait_until_color('white')
```

SPIKE™ Prime LEGO® Education

Montando uma Base Motriz Avançada

<https://education.lego.com/pt-br/lessons/prime-competition-ready/assembling-an-advanced-driving-base>



```
from spike import PrimeHub, MotorPair
from spike.control import wait_for_seconds, wait_until
from spike.operator import greater_than, less_than

hub = PrimeHub()
drive_motors = MotorPair('A', 'E')

drive_motors.set_default_speed(50)
drive_motors.set_motor_rotation(27.63, 'cm')

wait_for_seconds(1)

drive_motors.move(20, 'cm')
drive_motors.move(-20, 'cm')

drive_motors.move(20, 'cm', steering=-40)

hub.motion_sensor.reset_yaw_angle()

drive_motors.start(steering=100)
wait_until(hub.motion_sensor.get_yaw_angle, greater_than, 90)
drive_motors.stop()

drive_motors.start(steering=-100)
wait_until(hub.motion_sensor.get_yaw_angle, less_than, 0)
drive_motors.stop()
```

SPIKE™ Prime LEGO® Education

Meu Código, Nosso Programa

<https://education.lego.com/pt-br/lessons/prime-competition-ready/my-code-our-program>



```
from spike import PrimeHub, MotorPair
from spike.control import wait_for_seconds

hub = PrimeHub()
drive_motors = MotorPair('A', 'E')

drive_motors.set_default_speed(50)
drive_motors.set_motor_rotation(27.63, 'cm')

def square():
    for x in range(4):
        drive_motors.move(1.5, 'rotations')
        drive_motors.move(0.365, 'rotations', steering=100)

def triangle():
    for x in range(3):
        drive_motors.move(1.5, 'rotations')
        drive_motors.move(0.486, 'rotations', steering=100)

def circle():
    drive_motors.move(3, 'rotations', steering=60)

wait_for_seconds(1)

square()
hub.speaker.beep()

triangle()
hub.speaker.beep()

circle()
hub.speaker.beep()
```


SPIKE™ Prime LEGO® Education

Hora do Aprimoramento

<https://education.lego.com/pt-br/lessons/prime-competition-ready/time-for-an-upgrade>



```

from spike import PrimeHub, Motor

hub = PrimeHub()
lift_arm_motor = Motor('D')
dozer_blade_motor = Motor('C')

lift_arm_motor.set_default_speed(-100)
lift_arm_motor.run_for_seconds(1)
dozer_blade_motor.set_default_speed(-100)
dozer_blade_motor.run_for_seconds(1)

lift_arm_motor.set_default_speed(100)
lift_arm_motor.run_for_degrees(70)
dozer_blade_motor.set_default_speed(100)
dozer_blade_motor.run_for_degrees(70)
hub.beep()

lift_arm_motor.run_for_degrees(180)
lift_arm_motor.run_for_degrees(-180)
dozer_blade_motor.run_for_degrees(180)
dozer_blade_motor.run_for_degrees(-180)
hub.beep()

lift_arm_motor.run_for_degrees(180, speed=15)
lift_arm_motor.run_for_degrees(-180, speed=15)
dozer_blade_motor.run_for_degrees(180, speed=15)
dozer_blade_motor.run_for_degrees(-180, speed=15)

```

SPIKE™ Prime LEGO® Education

Missão Pronta

<https://education.lego.com/pt-br/lessons/prime-competition-ready/mission-ready>



```

from spike import Motor, MotorPair
from spike.control import wait_for_seconds

dozer_blade_motor = Motor('C')
lift_arm_motor = Motor('D')
drive_motors = MotorPair('A', 'E')

drive_motors.set_default_speed(25)
drive_motors.set_motor_rotation(27.63, 'cm')

dozer_blade_motor.start(-100)
lift_arm_motor.start(-100)
wait_for_seconds(1)
dozer_blade_motor.stop()
lift_arm_motor.stop()

dozer_blade_motor.run_for_degrees(70, speed=100)
lift_arm_motor.run_for_degrees(20, speed=100)

drive_motors.move(-2, 'cm')
drive_motors.move(10.5, 'cm')

dozer_blade_motor.run_for_degrees(180, speed=40)

drive_motors.move(-6, 'cm')

dozer_blade_motor.run_for_degrees(-180, speed=60)
dozer_blade_motor.run_for_degrees(180, speed=60)

drive_motors.move(7, 'cm')

dozer_blade_motor.run_for_degrees(-180, speed=60)

drive_motors.move(0.405, 'rotations', steering=-100)
drive_motors.move(60.5, 'cm', steering=-30)
drive_motors.move(34, 'cm')
drive_motors.move(32, 'cm', steering=-50)
drive_motors.move(17.5, 'cm')
drive_motors.move(0.415, 'rotations', steering=-100)
drive_motors.move(32, 'cm')

```

SPIKE™ Prime LEGO® Education

Passa a Peça

<https://education.lego.com/pt-br/lessons/prime-extra-resources/pass-the-brick>



```
from spike import Motor, PrimeHub
```

```
hub = PrimeHub()
grabber_motor = Motor('F')
```

```
# Isso fará a mão abrir uma vez para iniciar.
grabber_motor.run_for_seconds(1)
```

```
while True:
```

```
    # Isso fará a mão fechar quando você estiver pressionando o botão esquerdo do Hub.
    hub.left_button.wait_until_pressed()
    grabber_motor.set_stall_detection(False)
    grabber_motor.start(-75)
```

```
    # Isso fará a mão abrir quando você soltar o botão esquerdo do Hub.
    hub.left_button.wait_until_released()
    grabber_motor.set_stall_detection(True)
    grabber_motor.start(75)
```

SPIKE™ Prime LEGO® Education

Ideias, o jeito LEGO!

<https://education.lego.com/pt-br/lessons/prime-extra-resources/ideas-the-lego-way>



```
from spike import PrimeHub
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
```

```
while True:
```

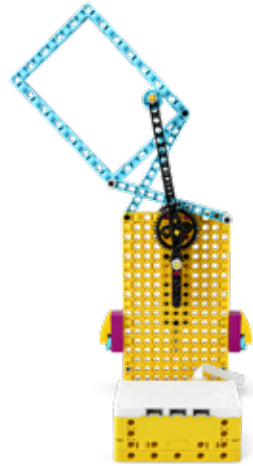
```
    if hub.left_button.was_pressed():
        hub.light_matrix.write('3')
        wait_for_seconds(1)
        hub.light_matrix.write('2')
        wait_for_seconds(1)
        hub.light_matrix.write('1')
        wait_for_seconds(1)
        hub.light_matrix.off()
        hub.speaker.beep(60, 0.5)
        hub.speaker.beep(72, 0.5)
```

```
    if hub.right_button.was_pressed():
        hub.light_matrix.write('5')
        wait_for_seconds(60)
        hub.light_matrix.write('4')
        wait_for_seconds(60)
        hub.light_matrix.write('3')
        wait_for_seconds(60)
        hub.light_matrix.write('2')
        wait_for_seconds(60)
        hub.light_matrix.write('1')
        wait_for_seconds(60)
        hub.light_matrix.off()
        hub.speaker.beep(60, 0.5)
        hub.speaker.beep(72, 0.5)
```

SPIKE™ Prime LEGO® Education

O que é isso?

<https://education.lego.com/pt-br/lessons/prime-extra-resources/what-is-this>



```
from spike import Motor

motor = Motor('F')

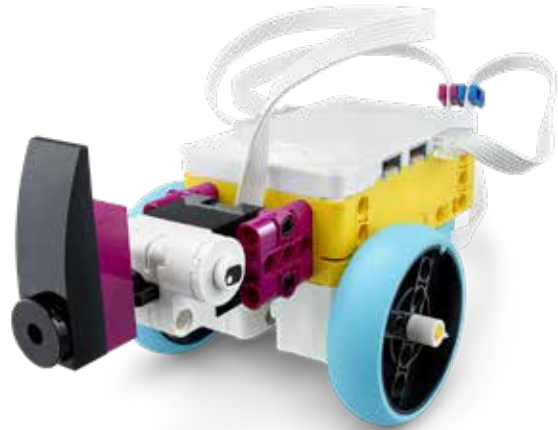
motor.set_stall_detection(False)

for x in range(5):
    motor.set_default_speed(50)
    motor.run_for_seconds(2)
    motor.set_default_speed(-50)
    motor.run_for_seconds(2)
```

SPIKE™ Prime LEGO® Education

Percorrendo a Distância

<https://education.lego.com/pt-br/lessons/prime-extra-resources/going-the-distance>



```
from spike import MotorPair

drive_motors = MotorPair('B', 'A')

drive_motors.set_default_speed(50)

drive_motors.move(10, 'rotations')
drive_motors.stop()
```

SPIKE™ Prime LEGO® Education

Go!

<https://education.lego.com/pt-br/lessons/prime-extra-resources/goal>



```
from spike import PrimeHub, Motor
from spike.control import wait_for_seconds
```

```
hub = PrimeHub()
kicker = Motor('A')
kicker.set_default_speed(100)
```

```
while True:
    kicker.run_to_position(0)

    hub.left_button.wait_until_pressed()
    kicker.run_for_rotations(1)
    wait_for_seconds(1)
```