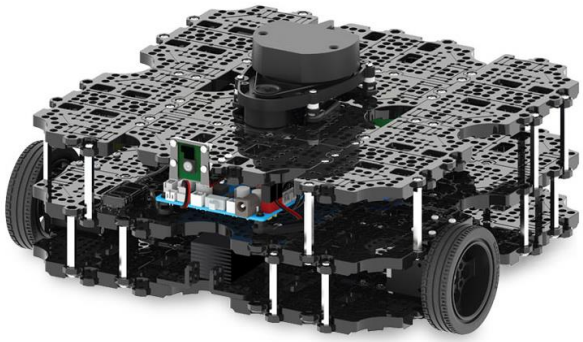
Mobile Robotics

AI camp



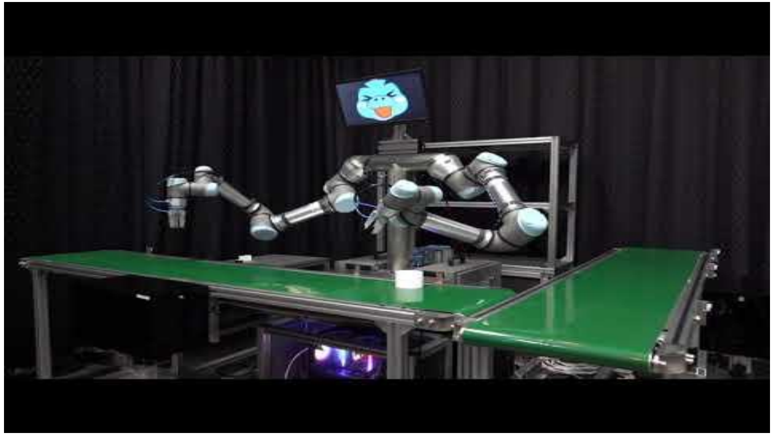
KVIS
KAMNOETVIDYA
SCIENCE ACADEMY

BRAIN
VISTEC

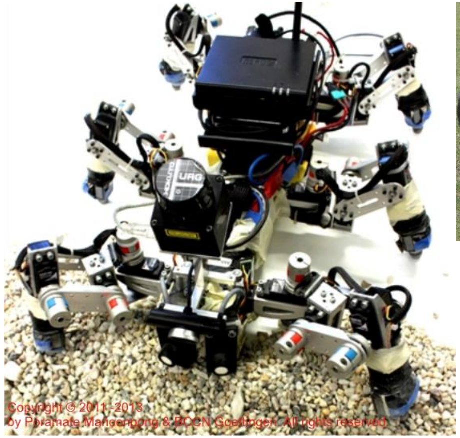
VISTEC
VIDYASIRIMEDHI
INSTITUTE OF SCIENCE AND TECHNOLOGY

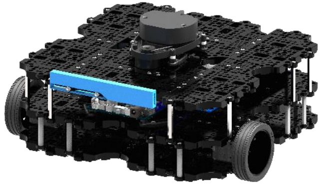
Robot

Stationary Robot



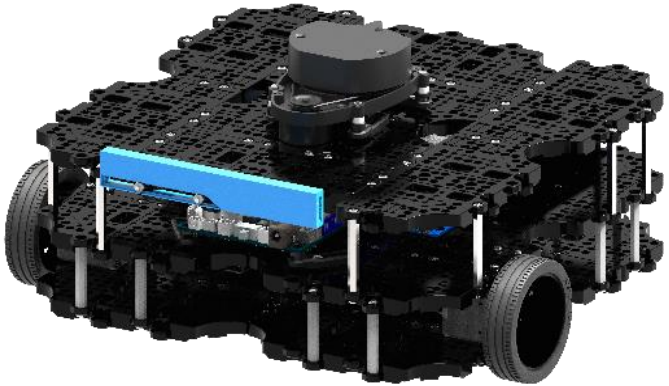
Mobile Robot





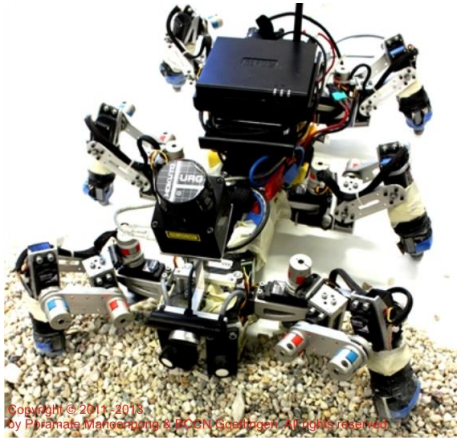
Mobile Robot

Wheeled Robot



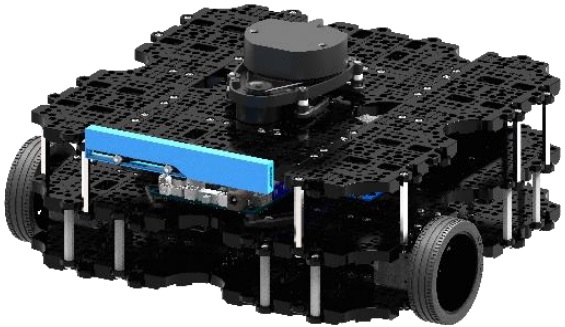
Track Robot

Legged Robot



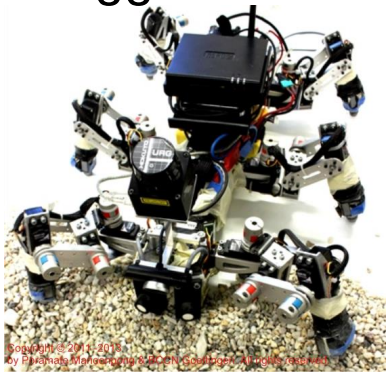
Mobile Robot

Wheeled Robot



Track Robot

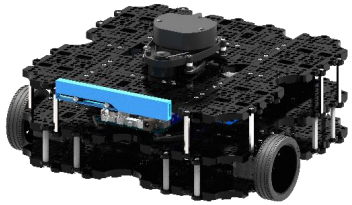
Legged Robot



	Wheeled Robot	Track Robot	Legged Robot
Flexibility	Normal	Low	High
Speed	High	Low	Low
Stability	Normal	High	Low
Complexity	Low	Low	High
Mobility	Low	Normal	High

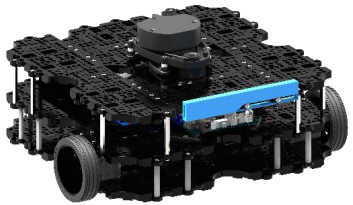
Programming

```
1  import time
2  from api import Turtlebot
3
4  robot = Turtlebot()
5  print "bringup finished"
6
7  robot.forward(50)  # 50% speed
8  time.sleep(3)
9  print "forward finished"
10
11 robot.backward(50)
12 time.sleep(3)
13 print "backward finished"
14
```



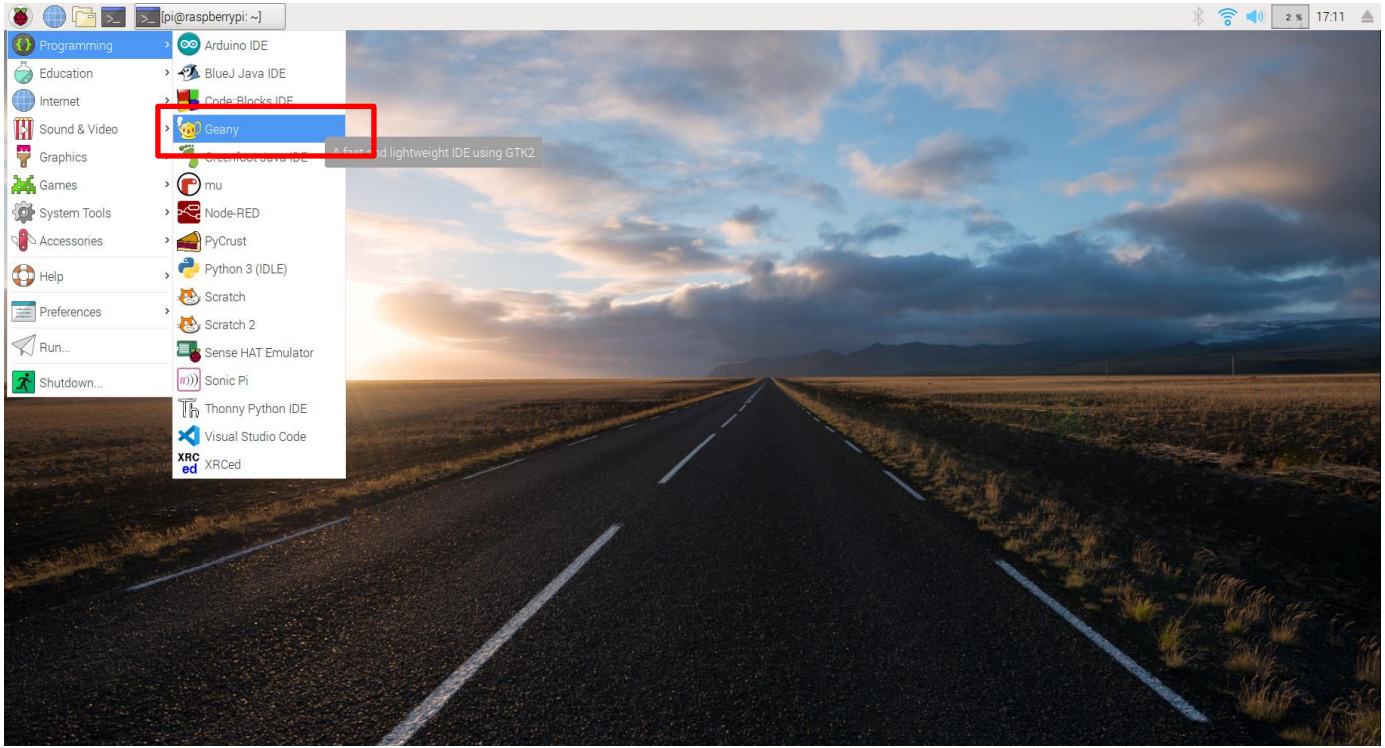
Programming

```
import time
from api import Turtlebot

robot = Turtlebot()
print "bringup finished"

robot.forward(50)  # 50% speed
time.sleep(3)
print "forward finished"

robot.backward(50)
time.sleep(3)
print "backward finished"
```

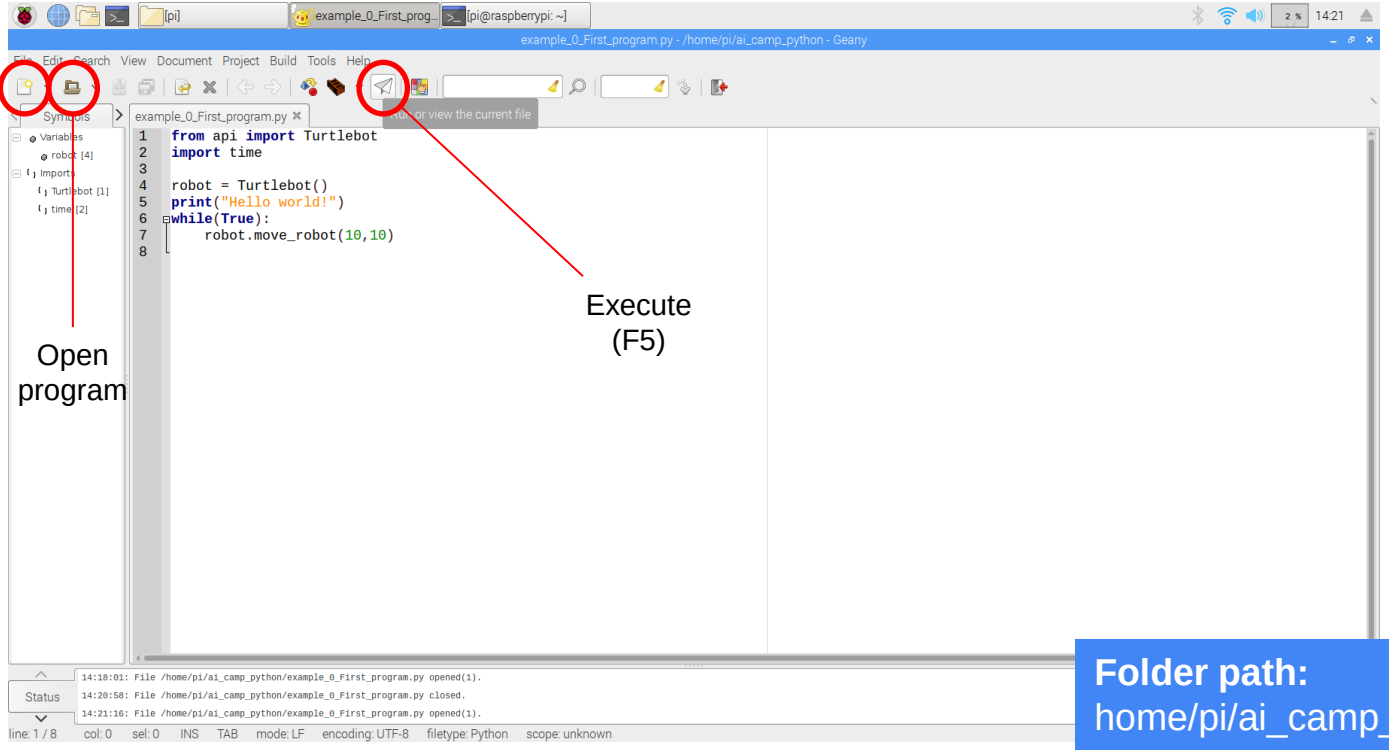



python™

Python editor “Geany”



Python editor “Geany”



Folder path:
home/pi/ai_camp_python/

Variable

Mathematics

$$y = \frac{x^2 + 3}{4}$$

Programming

```
1   x = 5
2   y = ((x ** 2) + 3) / 4.0
3   print(y)
4
```

7.0

Types of Variable

Depend on Programing language, Memory allocation, Hardware, etc.

The basic types are

- Integer
- Float
- String
- Boolean

```
1    a = 25
2    b = 2.5
3    c = "Hello"
4    d = True
```

How to declare variable?

Variable's Name = Value


```
1  a = 25
2  b = 2.5
3  c = "Hello"
4  d = True
5
6  ant = a / 5
7  bird = a / 5.0
8  cat = c + " World"
9  dog = a < b
```

Reserved word!!!

and	elif	if	print
as	else	import	raise
assert	except	in	return
class	finally	lambda	while
continue	for	not	with
def	from	or	yield
del	global	pass	

You can see this list any time by typing *help("keywords")*

Operator

Arithmetic Operators

Operator	Name	Example
+	Addition	x + y
-	Subtraction	x - y
*	Multiplication	x * y
/	Division	x / y
%	Modulus	x % y
**	Exponentiation	x ** y
//	Floor division	x // y

```
1  a = 25
2  b = 2.5
3  c = "Hello"
4  d = True
5
6  ant = a / 5
7  bird = a / 5.0
8  cat = c + " World"
9  dog = a < b
```

https://www.w3schools.com/python/python_operators.asp

Operator

Comparison Operators

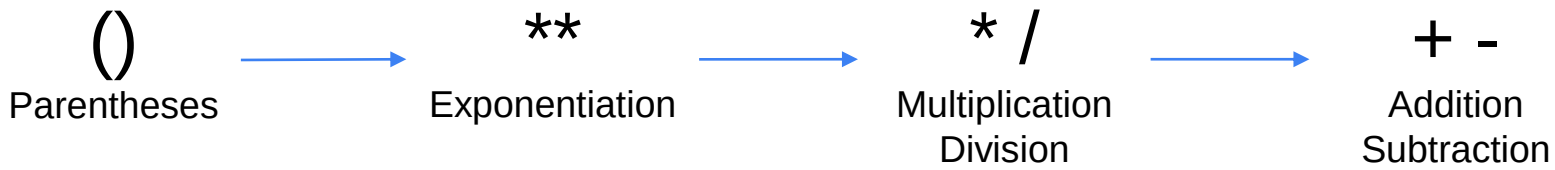
Operator	Name	Example
==	Equal	x == y
!=	Not equal	x != y
>	Greater than	x > y
<	Less than	x < y
>=	Greater than or equal to	x >= y
<=	Less than or equal to	x <= y

Logic Operators

Operator	Name	Example
and	Returns True if both statements are true	x < 5 and x < 10
or	Returns True if one of the statements is true	x < 5 or x < 4
not	Reverse the result, returns False if the result is true	not(x < 5 and x < 10)

https://www.w3schools.com/python/python_operators.asp

Rule of precedence



Same precedence
Do left to right

2 * (3-1) = **4**

(1+1)**(5-2) = **8**

2**1+1 = **3**

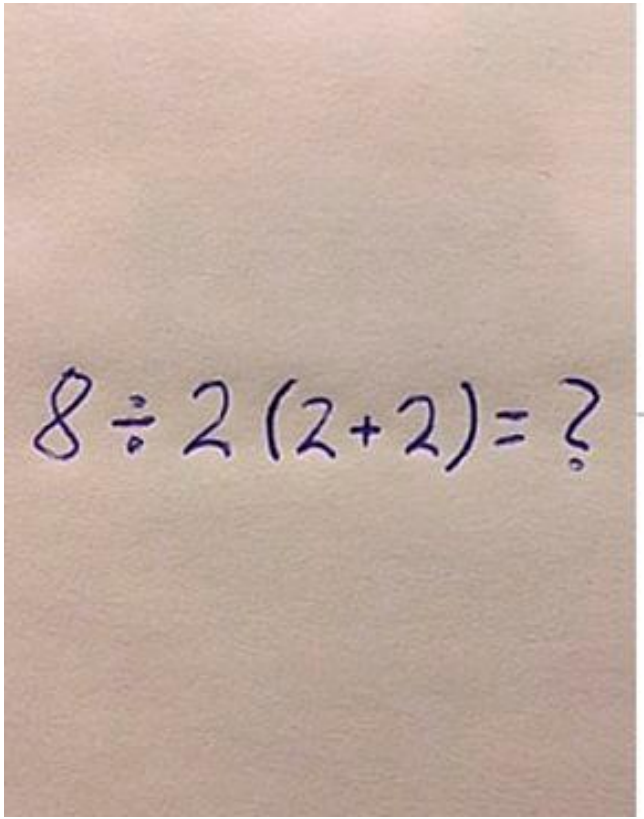
3*1**3 = **3**

2*3-1 = **5**

5-2*2 = **1**

6-3+2 = **5**

6-(3+2) = **1**



jake chyllenhaal
@corynhendrix

Took 3 calc classes, it's 16 bro

laur♏
@lauram_williams

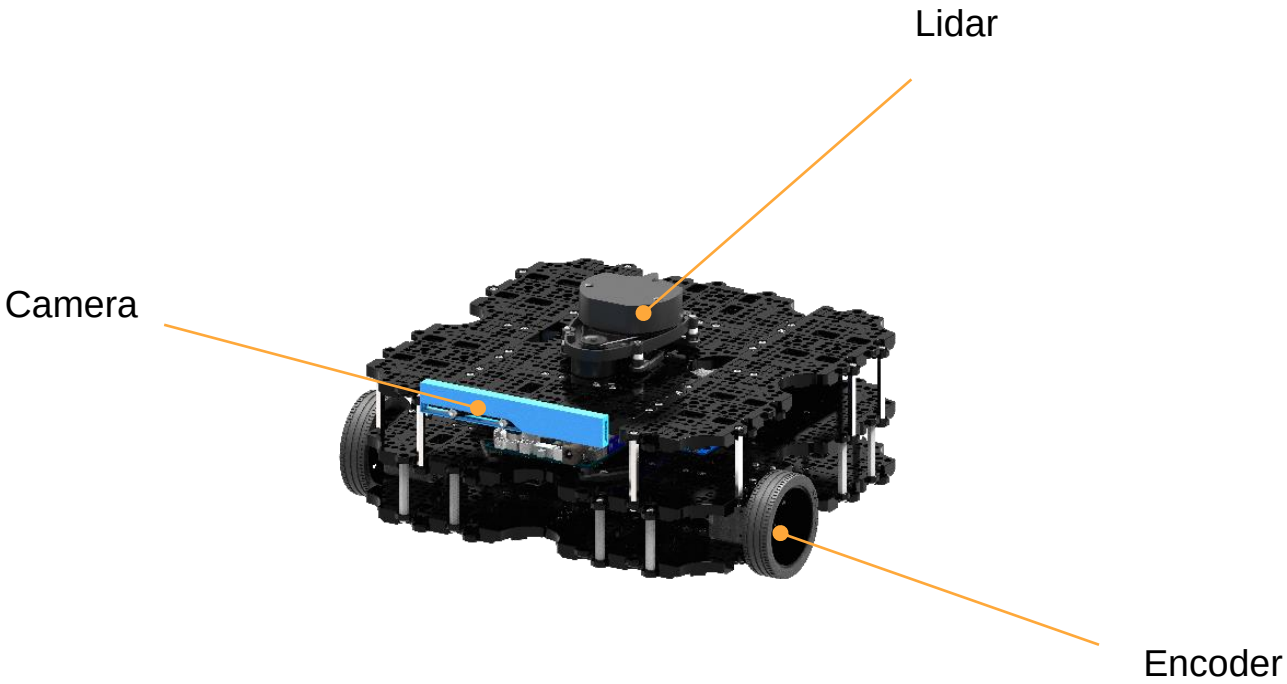
I have 2 math degrees it’s 1

Output

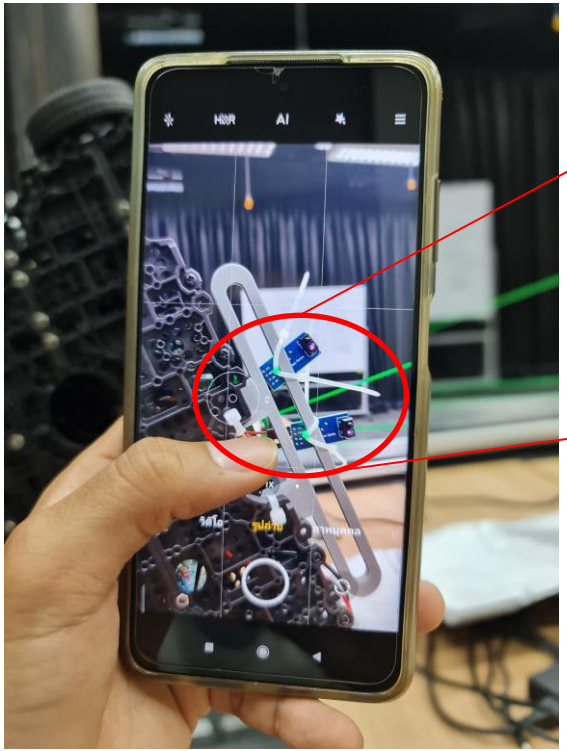
We use ***print()*** function to show the output

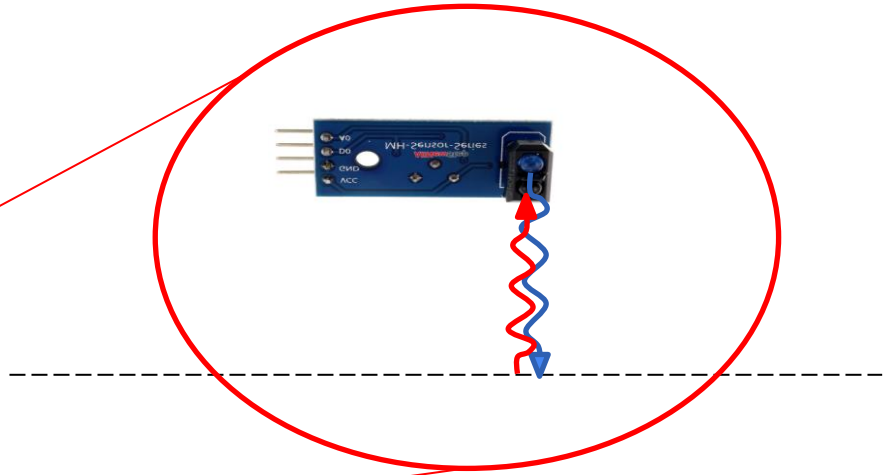
```
1  name = "Taluang"
2  year = 1980
3  animals = 500
4
5  # <-- This is called 'comment', everything after # symbol will not be run
6  print("Hello World!!")
7
8  # String must plus with String
9  print("This is " + name + " Zoo")
10
11 # str() function converts any variable type to  String type
12 print("This zoo has opened since " + str(year) + "s")
13
14 # You can directly pass any type of variable into print() function
15 print("The number of animals in this zoo is")
16 print(animals)
17
18 # You can print many variables at the same time by using , symbol
19 print("For example :")
20 print(200,"Dragons",140,"Mammoths",50,"Ostrich")
```

Input



IR sensor





<u>TCRT5000</u> IR Infrared Line Track Follower Sensor Obstacle Avoidance Module For Arduino

High reflexibility = Low sensor data
White < Black

Read IR data

Sensor_0

variable = **robot.get_sensor(Number_sensor)**

Sensor_1

Must write!!

```
1 from api import Turtlebot
2 import time
3
4 robot = Turtlebot()
5
6 ir_left = robot.get_sensor(0)
7 ir_right = robot.get_sensor(1)
8 ir_mid_left = robot.get_sensor(2)
9 ir_mid_right = robot.get_sensor(3)
```

Ready for coding!

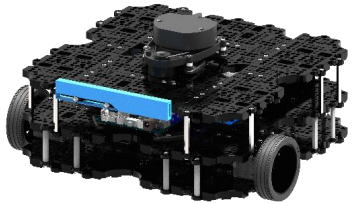
Quiz 1

- Read IR data, store it in the variable and print out
- check IR value of black and white surface

Condition

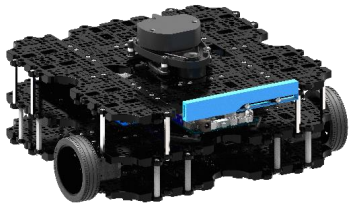
From the previous chapter

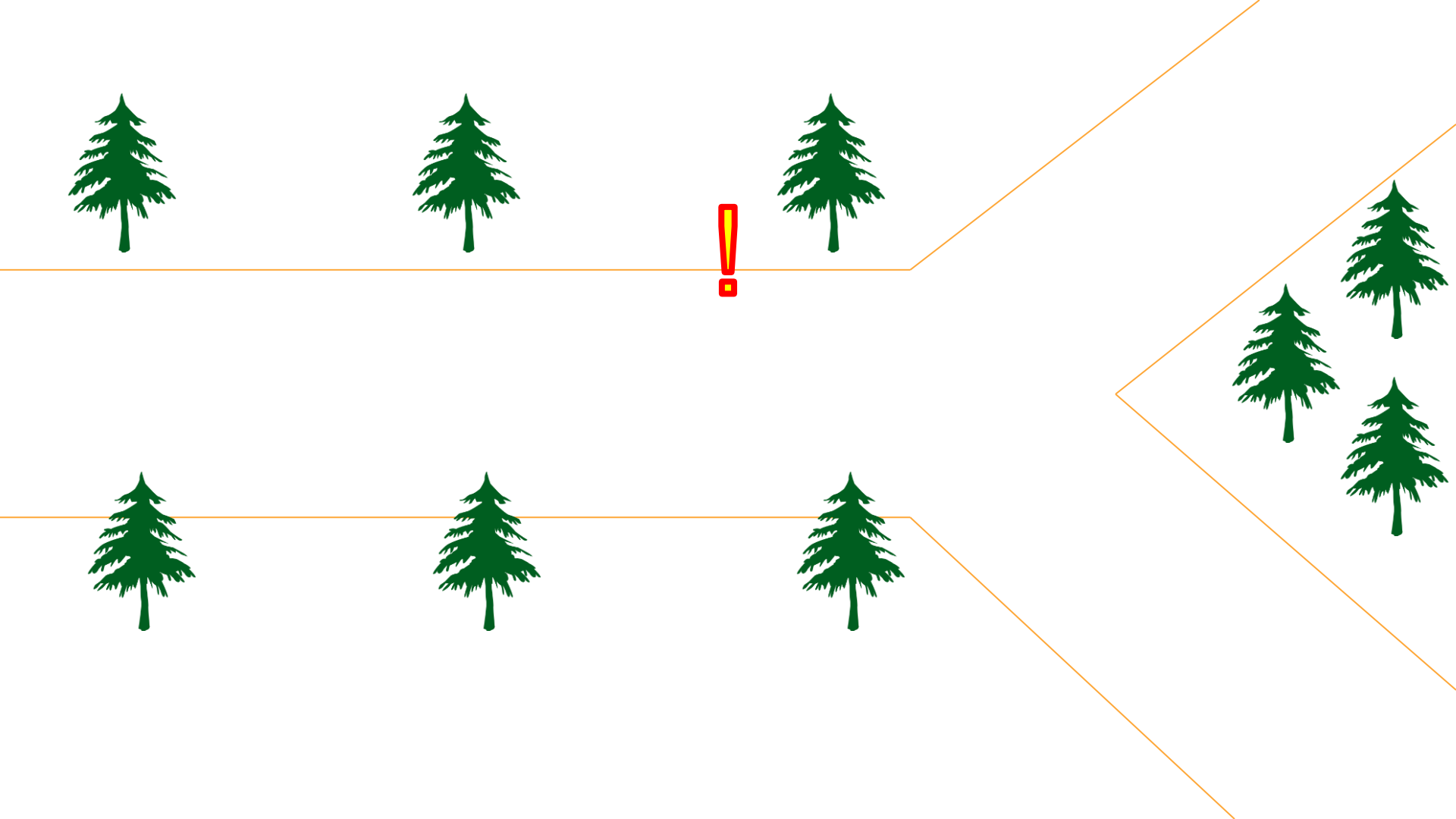
```
1  import time
2  from api import Turtlebot
3
4  robot = Turtlebot()
5  print "bringup finished"
6
7  robot.forward(50)  # 50% speed
8  time.sleep(3)
9  print "forward finished"
10
11 robot.backward(50)
12 time.sleep(3)
13 print "backward finished"
14
```



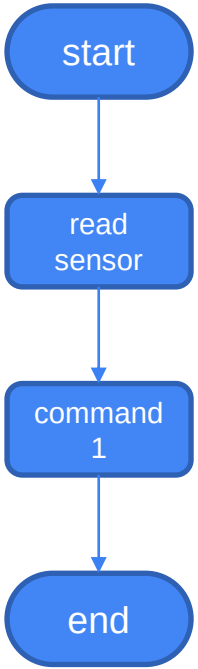
From the previous chapter

```
1  import time
2  from api import Turtlebot
3
4  robot = Turtlebot()
5  print "bringup finished"
6
7  robot.forward(50)  # 50% speed
8  time.sleep(3)
9  print "forward finished"
10
11 robot.backward(50)
12 time.sleep(3)
13 print "backward finished"
14
```

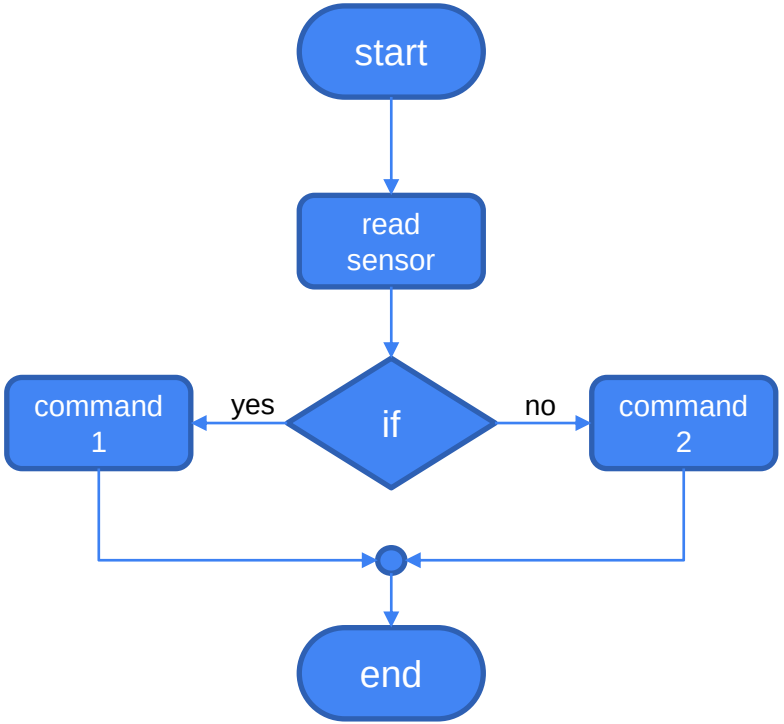




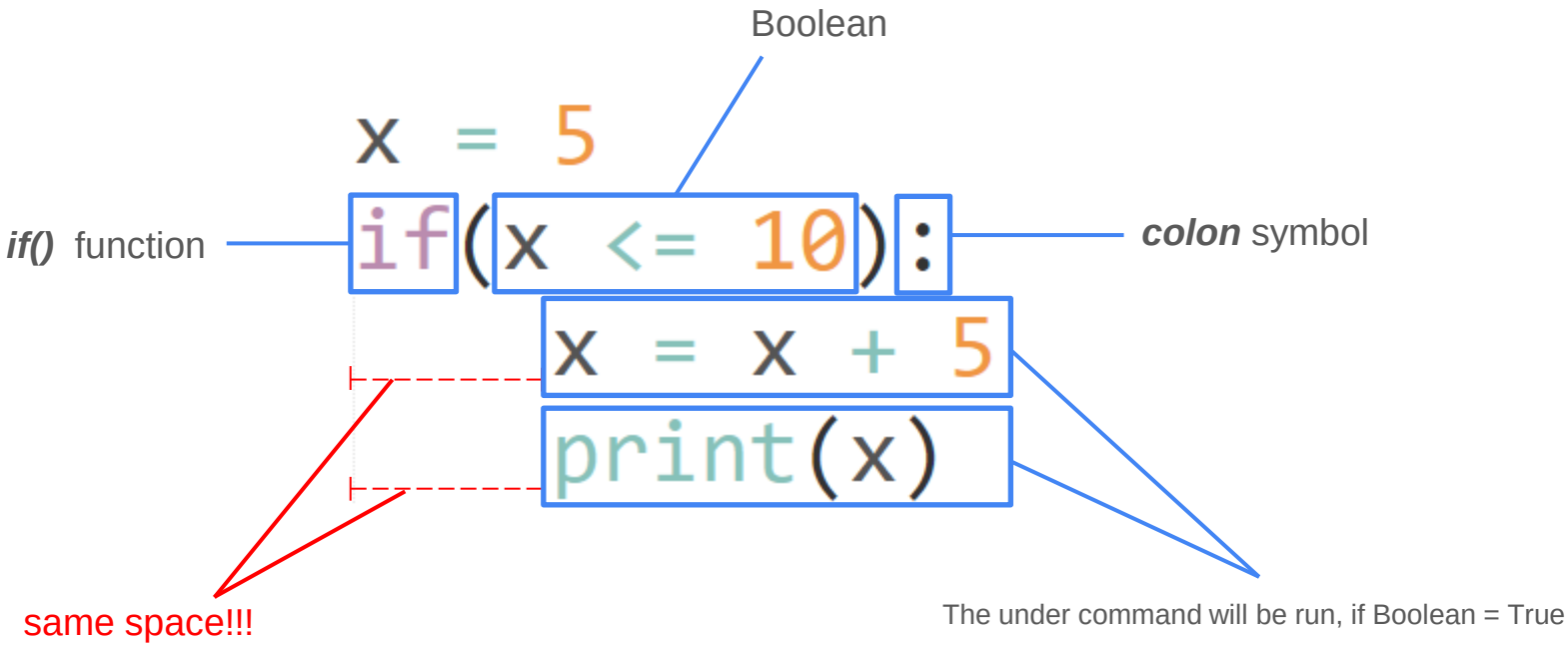
Sequence



Sequence with Decision



How to write “if”



How to write “if”

```
1   x = 5
2   if(x < 5):
3       x = x + 5
4
5   # you should add one enter after if statement for cleanly code
6   if(x < 3 or x > 0):
7       x = x - 3
8
9   if(x == 2 and not True):
10      x = x * 2
11
12  x = x - 1
13  print(x)
14
```

How to write “elif”

use ***if()*** with ***elif()*** to choose only one condition to be run

```
1  x = 5
2  if(x == 10):
3      print("Text_1 won't be show")
4  elif(x == 8):
5      print("Text_2 won't be show")
6  elif(x == 5):
7      print("But Text_3 will be show")
8  elif(x > 0):
9      print("Text_4 won't be show")
```

This command won’t be run!!!

How to write “else”

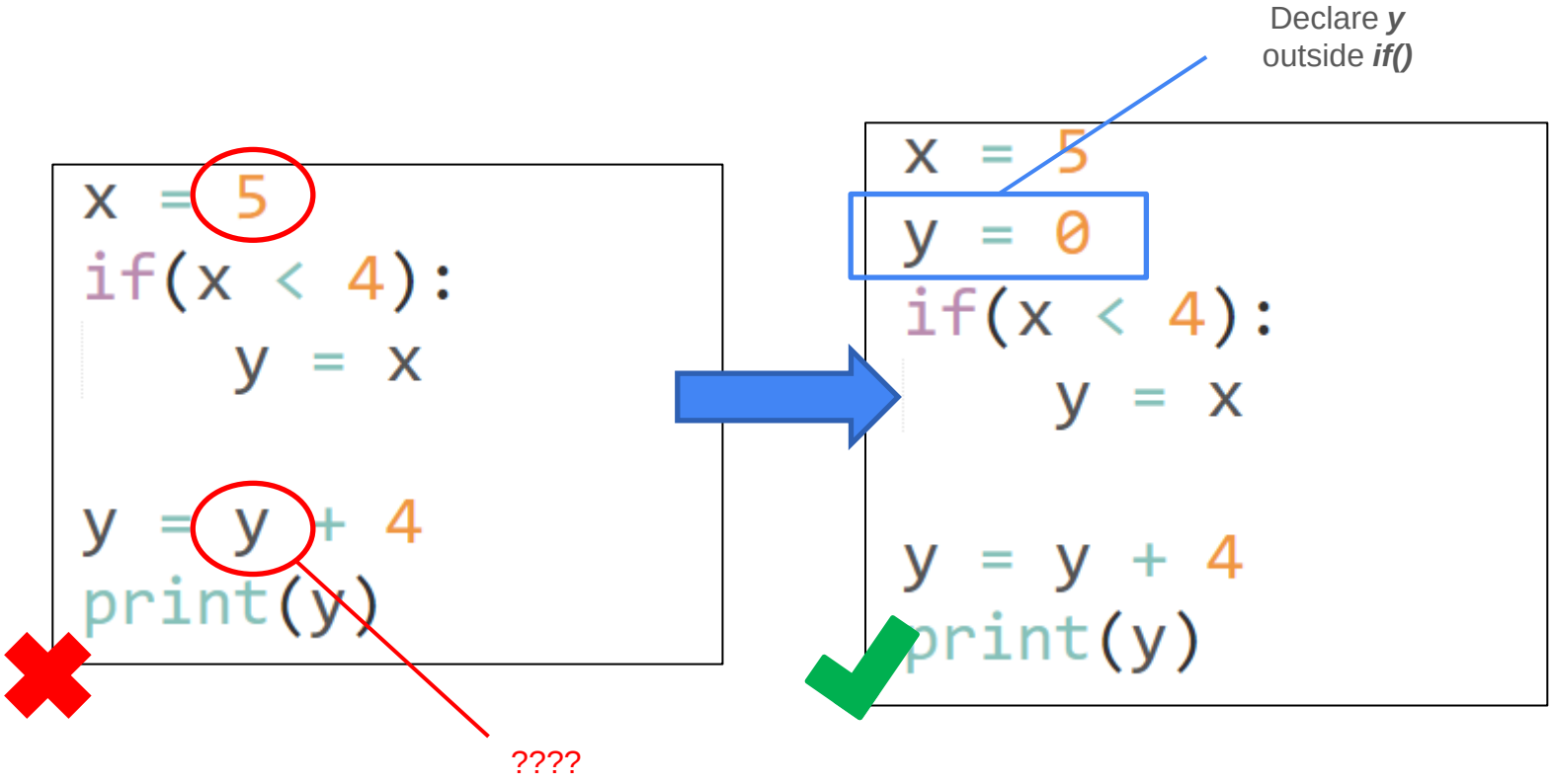
```
1 x = 5
2 if(x == 10):
3     print("Text_1 won't be show")
4 elif(x == 8):
5     print("Text_2 won't be show")
6 else:
7     print("But Text_3 will be show")
```

If no condition is True,
the command under ***else()*** will be run

Beware local variable!!!

```
x = 2
if(x < 4):
    y = x

y = y + 4
print(y)
```

Beware local variable!!!



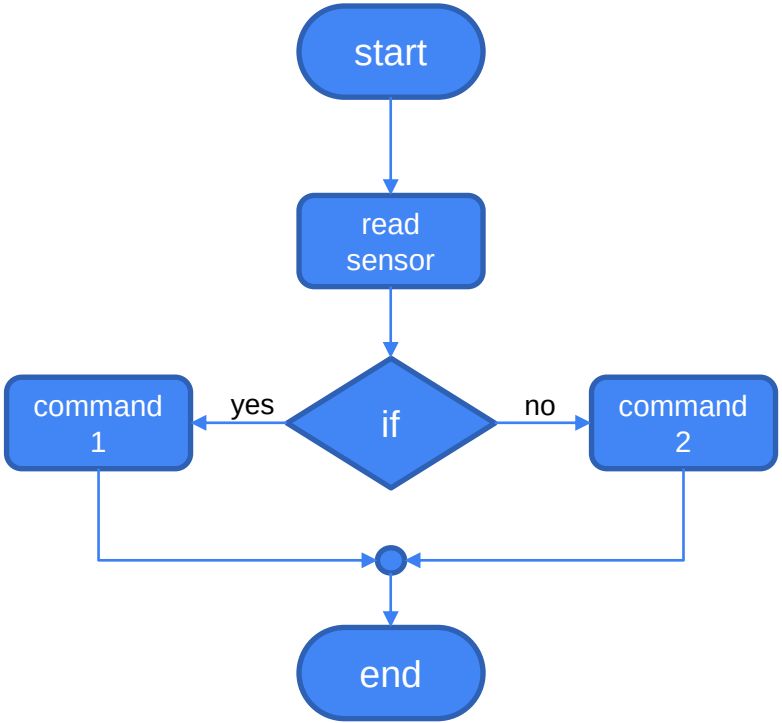
Quiz Time!

Quiz 2

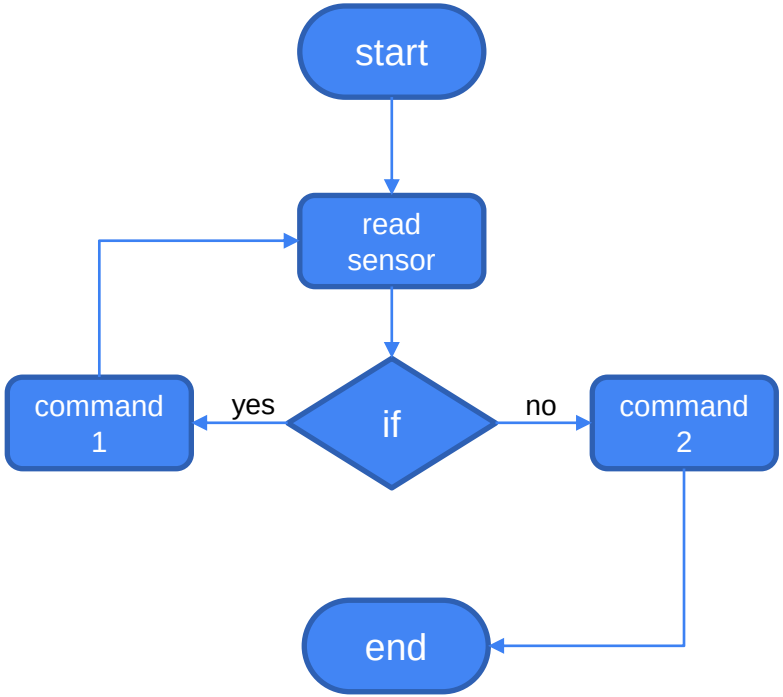
- Read IR data. If white surface, print white. If black surface, print black.

Loop

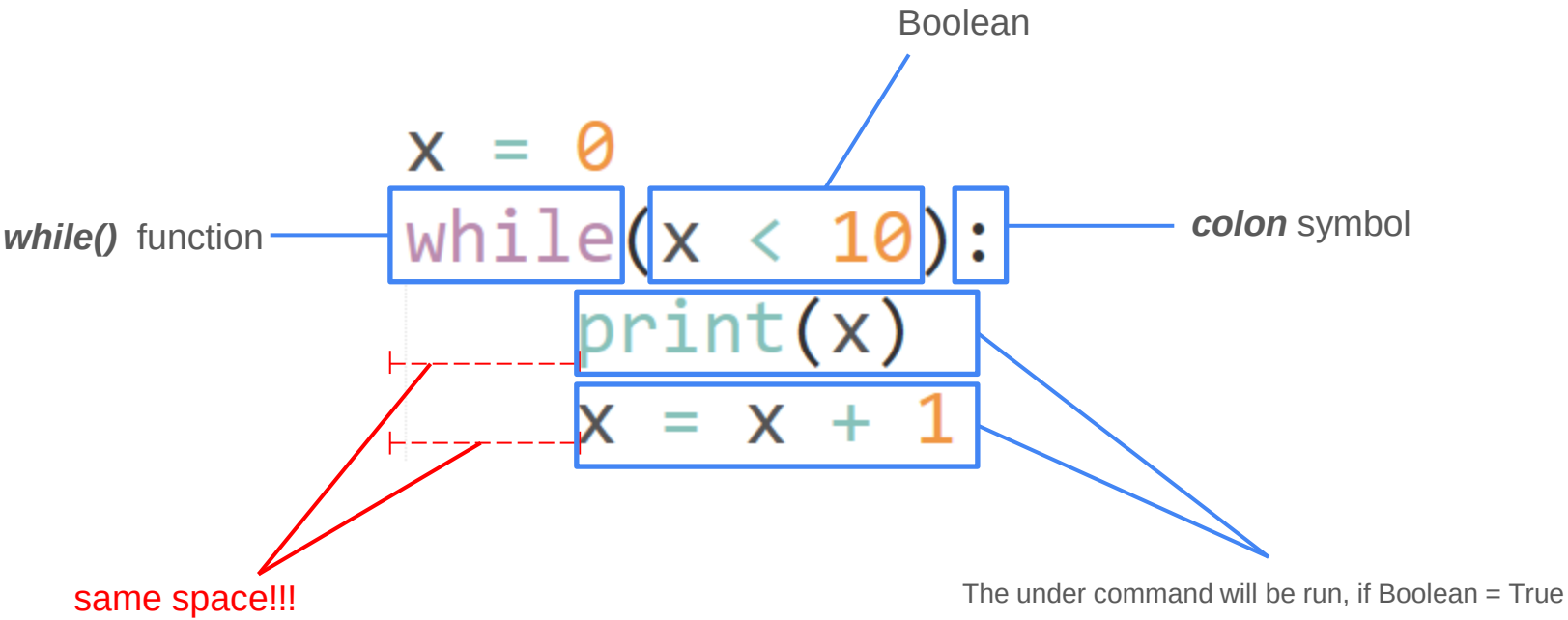
Decision



Decision with loop



How to write “while”



break / continue

use ***break*** to go out of loop

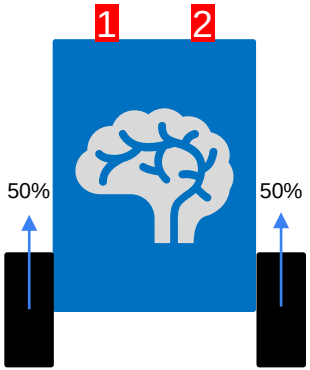
```
1  x = 0
2  while(x < 10):
3      if(x % 2 != 0):
4          x = x + 1
5          break
6
7      print(x)
8      x = x + 1
```

use ***continue*** to skip and go to next round

```
x = 0
while(x < 10):
    if(x % 2 != 0):
        x = x + 1
        continue

    print(x)
    x = x + 1
```

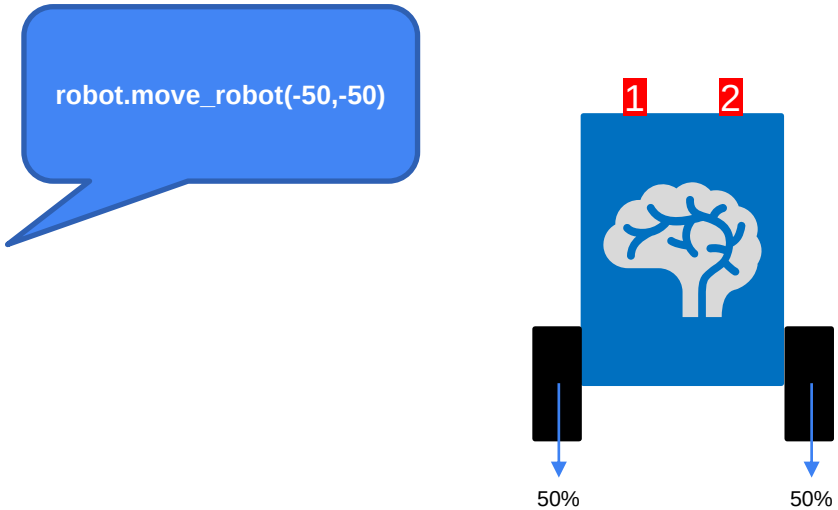
Motor control

robot.move_robot(%speed_motor_left, %speed_motor_right)



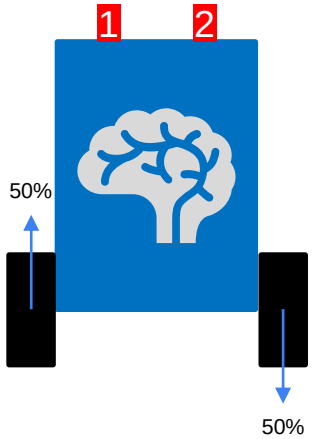
Motor control

robot.move_robot(%speed_motor_left, %speed_motor_right)



Motor control

robot.move_robot(%speed_motor_left, %speed_motor_right)



Motor control

robot.move_robot(0,0)
or
robot.stop()

Time delay

time.sleep(second)

Must write!!

```
1  from api import Turtlebot
2  import time
3
4  robot = Turtlebot()
5
6  robot.move_robot(50,50)
7  time.sleep(1)
8  robot.move_robot(-50,-50)
9  time.sleep(1)
10 robot.stop()
```

Loop with motor control

```
1  from api import Turtlebot
2  import time
3
4  robot = Turtlebot()
5  L_TH = 200
6  R_TH = 200
7
8  while(True):
9      ir_left = robot.get_sensor(0)
10     ir_right = robot.get_sensor(1)
11     print(ir_left,ir_right)
12     robot.move_robot(50,50)
13     if(ir_left>L_TH and ir_right>R_TH):
14         print("Stop")
15         break
```

Quiz Time!

Quiz 3

- Drive robot forward, stop at black line 3s , turn back180° and move forward 3s. (10 min)

Challenge!!!

Line Tracking

Line Tracking (Basic Turning Technique)

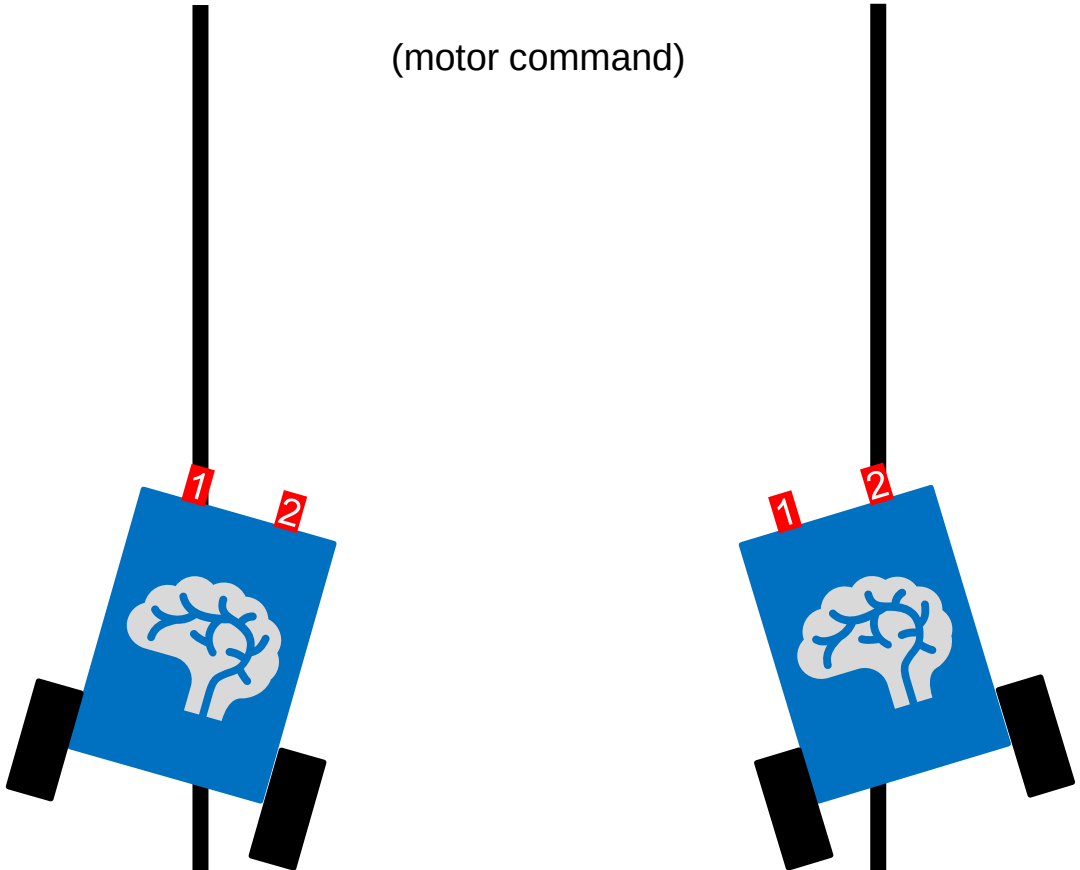
(motor command)

Line Tracking (Basic Turning Technique)

(motor command)

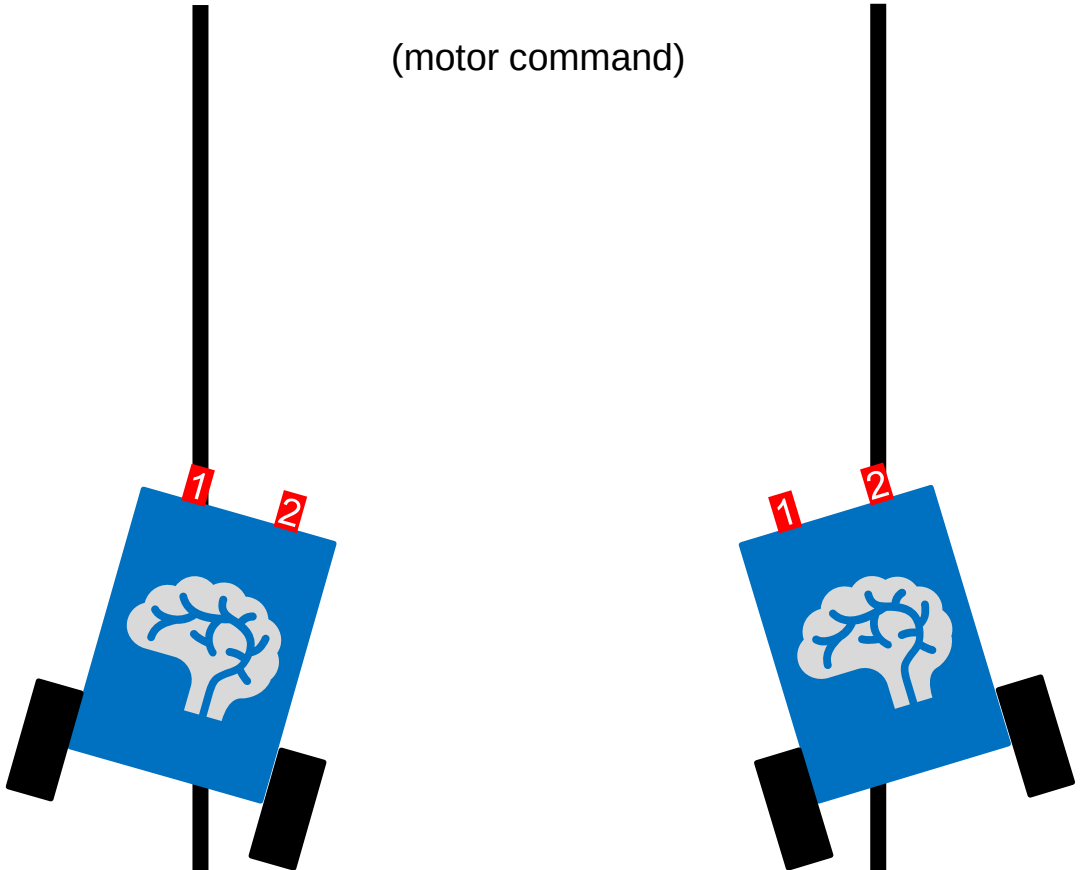
Line Tracking (Basic Line Tracking Technique)

(motor command)

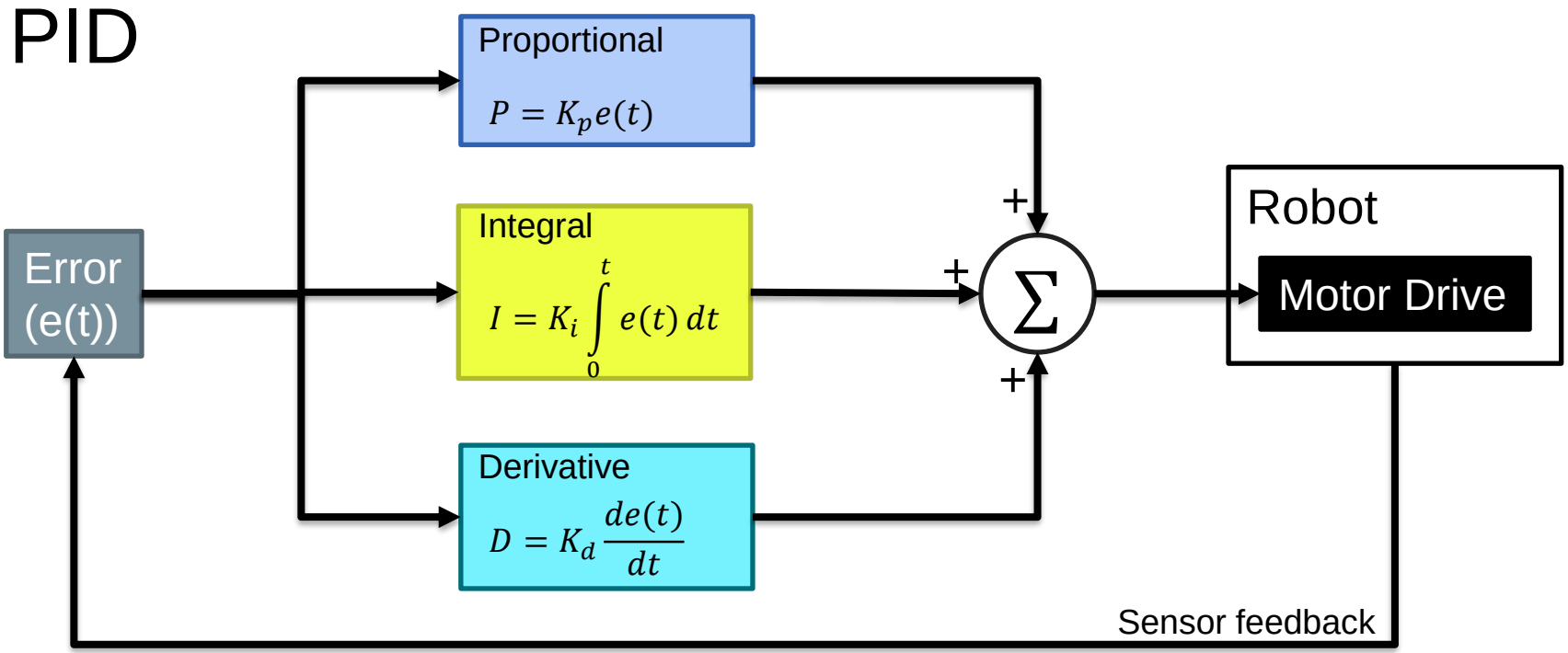


Line Tracking (Basic Line Tracking Technique)

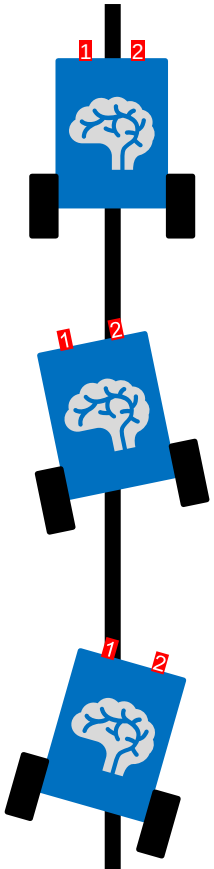
(motor command)



Line Tracking (Alternative Line Tracking Technique)



Line Tracking (Alternative Line Tracking Technique)



	Left motor	Right motor
	error = 0	error = 0
	error = 1	error = -1
	error = -1	error = 1

$P = K_p e(t)$ → P = Kp * error

$I = K_i \int_0^t e(t)\,dt$ → e_i = e_i + error
I = Ki * e_i

$D = K_d \frac{de(t)}{dt}$ → D = Kd * (error – error_p)

Motor = 20 + P+I+D

Line Tracking (Alternative Line Tracking Technique)

Bio-inspired