



## 6.5 Following a robot

Seeing more · Robot club · about 20 min

Name \_\_\_\_\_

Class \_\_\_\_\_

Date \_\_\_\_\_

### What this lesson is about

Two controllers that never finish: chase a moving tag at a set distance.

- Watch it go
- Two controllers, forever
- Corners
- Losing it

### Questions

6 marks in all

1. What does this program print?

[1 mark]

```
for dist in [50, 30, 20, 12]:  
    speed = max(0, min(70, (dist - 20) * 4))  
    print(dist, speed)
```

2. Why is the lowest speed in the follower 0, the `max(0, ...)`?

[1 mark]

- ☐ A So the robot stops when too close instead of reversing
- ☐ B So it never goes faster than the leader
- ☐ C So it keeps moving when the leader stops
- ☐ D Because negative speeds are an error

3. With `speed = max(0, min(70, (dist - 20) * 4))`, what speed does the follower use when the leader is 26 cm away?

[1 mark]

4. How is following a robot different from docking at a marker?

[1 mark]

- ☐ A The target keeps moving, so the loop never finishes
- ☐ B It uses a different detector
- ☐ C It needs no rotation controller
- ☐ D The distance is measured in pixels

5. When the leader goes out of view, the lesson just turns right. What is a better recovery?

[1 mark]

- ☐ A Remember the last bearing and turn that way
- ☐ B Stop and wait for the leader to come back
- ☐ C Drive forward at full speed
- ☐ D Switch to the line detector

6. The task starter always drives at 60 towards robot 101. What goes wrong?

[1 mark]

- ☐ A It rams the leader, because nothing slows it down as it gets close
- ☐ B It never catches up
- ☐ C It turns away from the leader
- ☐ D It loses the leader immediately

## The task: follow the leader

Stay 12 to 35 cm behind the green robot for at least 80% of the run, without touching it. The starter drives at the leader at a fixed speed and rams it. Add the distance controller.

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

# do this 390 times (tick counts from 0)
for tick in range(390):
    seen = [r for r in robots() if r[0] == "green"]
    if seen:
        # forward, sideways, rotation: -100 to 100 each, until the next command
        drive(60, 0, (seen[0][1] - 160) * 120 / 320 * 3)
        # pause 0.1 s (the robot keeps doing what it was told)
        wait(0.1)
    # all motors off
    stop()
```

Plan your program here, then type it in and press Run.



**Do it on the robot**

[www.bugbotlab.com/learn/6-5-following-a-robot/](http://www.bugbotlab.com/learn/6-5-following-a-robot/)

The simulator checks it and tells you when it passes. Nothing to install, no account.

## Challenges

1. Hold 30 cm instead of 20. How far behind does it sit on the straight now, and does it still stay inside 12 to 35 cm?
2. Print your distance to the leader every second and see how steady it is.
3. Make the LED green while you can see the leader and red when you cannot.