



0.2 Numbers and names

What this lesson is about

Variables, arithmetic, and driving an exact distance instead of counting seconds.

- A name for a number
- Driving an exact distance

Questions

7 marks in all

1. What does this program print?

[1 mark]

```
speed = 60
seconds = 1.5
print("drove at", speed, "for", seconds, "seconds")
```

Answer:

drove at 60 for 1.5 seconds

Once a variable has a value, its name stands for that value. print shows the values, not the names.

2. What does this program print?

[1 mark]

```
cm_per_second = 18
seconds = 2
print("that should be about", cm_per_second * seconds, "cm")
```

Answer:

that should be about 36 cm

* means times. Python works out 18 times 2 before printing it.

3. What does this program print?

[1 mark]

```
speed = 60
speed = 25
print("speed is", speed)
```

Answer:

speed is 25

A variable holds one value at a time. The second line replaces 60 with 25, so 25 is what gets printed.

4. What does `distance=20` mean in `forward(50, distance=20)`? [1 mark]
- ☐ A Drive 20 cm, then stop
 - ☐ B Drive for 20 seconds
 - ☐ C Drive at speed 20
 - ☐ D Drive 50 cm, 20 times

Answer: A. The 50 is the speed and `distance=20` is how far, in centimetres. Naming the argument makes it clear which number is which.

5. Do you need a `wait()` after `forward(50, distance=20)`? [1 mark]
- ☐ A No, the command waits until the robot has gone that far
 - ☐ B Yes, `wait(20)`, or the robot never goes anywhere
 - ☐ C Yes, otherwise it never stops
 - ☐ D Only when the speed is over 50

Answer: A. With a distance, the command itself waits until the move is done. `wait()` is for commands like `forward(50)` that return straight away.

6. What does this program print? [1 mark]

```
travelled_cm = 85
print("that is", travelled_cm / 100, "metres")
```

Answer:

that is 0.85 metres

/ divides. There are 100 centimetres in a metre, so 85 cm is 0.85 metres.

7. The robot covers 18 cm every second. How many centimetres does it cover in 2.5 seconds? Give the exact number. [1 mark]

Answer: 45. Distance is speed times time: 18 times 2.5 is 45 cm.

The task: two legs

Drive 40 cm forward, then 45 cm to the right, and stop in the green zone. Use `distance=` rather than counting seconds, and give the two distances names at the top.

```
from bugbot import *
connect()

first = 40
# your second distance here

forward(50, distance=first)
stop()
```

The hint students can ask for: Forward moves you up the mat, right moves you across it. Both take distance= in centimetres.

A solution

```
from bugbot import *
connect()
first = 40
second = 45
forward(50, distance=first)
right(50, distance=second)
stop()
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.