



5.1 The loop

What this lesson is about

Look, decide, act, wait, repeat. One action per tick. Counting ticks.

- Scripts block; behaviours tick
- One action per tick
- Why the wait matters
- Counting ticks

Questions

7 marks in all

1. What does every behaviour do, again and again? [1 mark]

- ☐ A Look, decide, act, wait, repeat
- ☐ B Drive, stop, turn, stop
- ☐ C Wait until the task is finished, then end
- ☐ D Read every sensor once, then act forever

Answer: A. A behaviour never finishes a move before looking again. Each tick it looks, picks one action and waits a tick.

2. Which call suits a behaviour loop, because it does not block? [1 mark]

- ☐ A `forward(60)`
- ☐ B `forward(60, distance=20)`
- ☐ C `turn_right(30, angle=90)`
- ☐ D `wait(5)`

Answer: A. `forward(60)` starts driving and returns at once. With a distance or angle, your program waits until the move is done.

3. A student leaves `wait(0.1)` out of a `while True` behaviour loop. What goes wrong? [1 mark]

- ☐ A The loop asks the sensors thousands of times with no time passing, and the simulator stops it
- ☐ B Nothing, it just runs faster
- ☐ C The robot only does the last tick
- ☐ D The robot drives backwards

Answer: A. The tick needs time to pass so the robot actually moves between looks. The simulator stops the program after 5,000 sensor reads with no time passing. On the real robot the radio would choke.

4. What does this program print?

[1 mark]

```
for tick in [0, 9, 10, 19, 20, 35]:  
    print(tick, "green" if tick % 20 < 10 else "off")
```

Answer:

```
0 green  
9 green  
10 off  
19 off  
20 green  
35 off
```

tick % 20 counts 0 to 19 and starts again. Under 10 is green, so it is green for the first second of every two. 35 % 20 is 15, so off.

5. What does this program print?

[1 mark]

```
def choose(d):  
    if d < 15:  
        return "back"  
    elif d < 30:  
        return "turn"  
    else:  
        return "forward"  
  
for d in [10, 15, 29, 30, 80]:  
    print(d, choose(d))
```

Answer:

```
10 back  
15 turn  
29 turn  
30 forward  
80 forward
```

Only the first true test counts. 15 is not under 15, so it falls to the turn. 30 is not under 30, so it goes forward.

6. In one tick a program calls `forward(60)` and then `turn_right(60)`. What does the robot do?

[1 mark]

- ☐ A Turns right: only the last command counts
- ☐ B Drives forward: only the first command counts
- ☐ C Drives forward and turns at once
- ☐ D Stops, because the commands clash

Answer: A. Each command replaces the one before. That is why one tick should choose exactly one action with `if / elif / else`.

7. A behaviour loop waits 0.1 s each tick. How many ticks is five seconds?

[1 mark]

Answer: 50. Ten ticks a second, so five seconds is 50 ticks.

The task: wander

Keep moving for the whole twenty seconds, at least 120 cm in total, without touching a box or the edge of the mat.

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

# drive forward at 60 (keeps going until the next command)
forward(60)
# pause 5 s (the robot keeps doing what it was told)
wait(5)
# all motors off
stop()
```

The hint students can ask for: Keep driving for the whole 20 seconds, at least 120 cm in total, without touching a box or the edge. Look, decide, act, wait, repeat.

A solution

```
from bugbot import *
connect()
for tick in range(190):
    if distance() < 25:
        turn_right(60)          # something close: turn away
    else:
        forward(60)             # clear: go
    wait(0.1)
stop()
```

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