



0.6 Lists of readings

What this lesson is about

Lists, len, min and max, slices and looping over the 64 readings of the depth grid.

- A list
- The depth grid
- Going through a list
- Where next

Questions

7 marks in all

1. What does this program print?

[1 mark]

```
speeds = [20, 40, 60]
print(speeds[0])
print(len(speeds))
```

Answer:

```
20
3
```

Counting starts at 0, so `speeds[0]` is the first item, 20. `len()` counts how many items there are: 3.

2. `speeds = [20, 40, 60]`. Which gives you 60?

[1 mark]

- ☐ A `speeds[2]`
- ☐ B `speeds[3]`
- ☐ C `speeds[1]`
- ☐ D `speeds(3)`

Answer: A. Positions count from 0, so the three items are at 0, 1 and 2. `speeds[3]` would be a fourth item, which does not exist.

3. How many readings does `tof_grid()` give you?

[1 mark]

Answer: 64. It is an 8 by 8 square of distances, and 8 times 8 is 64.

4. What does this program print?

[1 mark]

```
grid = [10, 11, 12, 13, 14, 15]
print(grid[2:4])
```

Answer:

```
[12, 13]
```

A slice goes from the first number up to but not including the second, so `[2:4]` is the items at positions 2 and 3.

5. What does this program print?

[1 mark]

```
readings = [55, 32, 18, 70, 39]
close = 0
for value in readings:
    if value < 40:
        close = close + 1
print(close, "readings are under 40 cm")
```

Answer:

3 readings are under 40 cm

The loop looks at each reading in turn. 32, 18 and 39 are under 40, so `close` goes up three times.

6. What does this program print?

[1 mark]

```
readings = [27.43, 80.0, 45.2]
print("nearest:", round(min(readings), 1))
```

Answer:

nearest: 27.4

`min()` picks the smallest item, 27.43, and `round(..., 1)` keeps one decimal place.

7. Which readings does `grid[24:32]` give you?

[1 mark]

- ☐ A The 8 readings at positions 24 to 31
- ☐ B The 9 readings at positions 24 to 32
- ☐ C Only readings 24 and 32
- ☐ D The 32 readings after position 24

Answer: A. A slice stops just before its second number, so `[24:32]` is 24, 25 and so on up to 31: eight readings, one row of the grid.

The task: how near is the nearest

There is a box on the mat in front of the robot. Print one line saying how far away it is, like `nearest: 27`, worked out with `min()` from `tof_grid()` rather than from `distance()`. The smallest of all 64 readings is the mat under the robot's nose, 8 cm, so only look at the rows that see straight ahead.

```
from bugbot import *
connect()

grid = tof_grid()
# your line here
```

The hint students can ask for: `tof_grid()` hands back 64 whole-centimetre numbers. The bottom rows see the mat, so take `min()` over the rows that look straight ahead (rows 2 and 3).

A solution

```
from bugbot import *
connect()
grid = tof_grid()
print('nearest:', min(grid[16:32]))
```

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