



0.6 Lists of readings

Name _____

Class _____

Date _____

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))
```

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

[1 mark]

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

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

[1 mark]

4. What does this program print?

[1 mark]

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

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")
```

6. What does this program print?

[1 mark]

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

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

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
```

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



Do it on the robot

www.bugbotlab.com/learn/start-6-lists/

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

Challenges

1. Print the furthest reading as well, on its own line.
2. Print how many readings are under 30 cm.
3. Turn the robot 45 degrees, take the readings again, and see which way is clearest.