



6.7 Where will it be

Seeing more · Robot club · about 20 min

What this lesson is about

Two sightings give a velocity: give way to a robot crossing your path.

- A robot that crosses your path
- Give way
- Predicting

Questions

7 marks in all

1. Robot 102's bearing sweeps from -36 to 33 degrees while its distance only goes from 58 cm down to 40 [1 mark]
cm and back out. What is it doing?

- ☐ A Crossing in front of you
- ☐ B Driving straight at you
- ☐ C Driving away from you
- ☐ D Standing still

Answer: A. A bearing sweeping through zero while the distance stays about the same is the pattern of a robot crossing your path.

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

```
last = None
for bearing in [-30.0, -20.0, -8.0, 5.0, 18.0]:
    if last is not None:
        print(bearing, "coming" if abs(bearing) < abs(last) else "leaving")
    last = bearing
```

Answer:

```
-20.0 coming
-8.0 coming
5.0 coming
18.0 leaving
```

Comparing sizes with `abs` ignores which side it is on. The bearing shrinks until it crosses the middle, then grows.

3. The first give way rule stops whenever robot 102 is close and in front. What is wrong with it? [1 mark]

- ☐ A It also stops for a robot that is already moving away from its path
- ☐ B It never stops at all
- ☐ C It drives into the robot
- ☐ D It only works on the left side

Answer: A. It is safe but too cautious. Only a robot heading for your path is a problem.

4. The rule is `tags` and `tags[0][3] < 30` and `abs(bearing_of(tags[0][1])) < 35`. Robot 102 is 25 [1 mark]
cm away at a bearing of -40. Does the robot stop?
- ☐ A No: 40 degrees is outside the 35 degree window
 - ☐ B Yes: it is closer than 30 cm
 - ☐ C Yes: the bearing is negative
 - ☐ D No: `abs` makes the bearing 0

Answer: A. Both tests must be true. It is close enough, but too far off to the side to count as in the way.

5. When robot 102 is out of view, the prediction code sets `last = None`. Why? [1 mark]
- ☐ A An old bearing from before it was lost would give a wrong comparison next time
 - ☐ B `None` makes the robot stop
 - ☐ C Otherwise the tag detector switches off
 - ☐ D To count how long it was lost

Answer: A. A comparison only means something between two sightings in a row. Starting fresh avoids a false alarm.

6. `waited` goes up by 1 on every tick the robot stops, at ten ticks a second. At the end `waited` is 23. [1 mark]
What does `waited / 10` print?

Answer: 2.3 (accept within 0.05). 23 ticks at a tenth of a second each is 2.3 seconds of waiting.

7. What does the general version of prediction, used by cars and ships, do? [1 mark]

Tick every answer that is true.

- ☐ A Keep the last few positions of the other robot
- ☐ B Work out its velocity from them
- ☐ C Check whether the two paths cross in the next second or two
- ☐ D Stop for every robot it can see

Answer: A, B, C. Position history gives velocity, and velocity tells you where it is about to be. Stopping for everything is what prediction avoids.

The task: give way

Reach the green zone without touching the red robot.

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

# drive forward at 50 for 80 cm, then stop
forward(50, distance=80)
```

The hint students can ask for: A robot with a red dome shuttles back and forth across your path. Watch where it is going, wait when it is about to cross in front of you, and reach the goal without touching it.

A solution

```
from bugbot import *
connect()
for tick in range(390):
    x, y = position()
    if y > 80:
        break
    seen = [r for r in robots() if r[0] == 'red']
    in_the_way = False
    if seen:
        cx, dist = seen[0][1], seen[0][3]
        bearing = (cx - 160) * 120 / 320
        if dist < 40 and abs(bearing) < 40:      # close and in front of me
            in_the_way = True
    if in_the_way:
        stop()                                  # give way: let it pass
    else:
        forward(50)
        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.