ANSWERS

10.4 Trails and tight spots

Competing · Robot club · about 15 min

What this lesson is about

A mat that fills up: seeing trails, turning early, keeping to the outside, and turning without stopping.

- Seeing a trail
- Never drive into a dead end

Questions

6 marks in all

1. In Light Cycles you want a moment to think. What happens if you stop? [1 mark]

- ☐ **A** You lose: sitting still is not allowed
- ☐ **B** Nothing, the trails pause too
- ☐ **C** Your trail disappears
- ☐ **D** The other robots stop as well

> **Answer:** A. Stopping to think is the same as losing. Turn while moving instead.

2. Which line turns a corner without stopping? [1 mark]

- ☐ **A** `drive(35, 0, 80)`
- ☐ **B** `stop()` then `turn_right(40, angle=90)`
- ☐ **C** `turn_right(40)`
- ☐ **D** `backward(30)` then `turn_left(40)`

> **Answer:** A. `drive(forward, sideways, turn)` sets all three at once, so you keep moving forward while turning.

3. Which of these keep you alive longer in Light Cycles? [1 mark]

Tick every answer that is true.

- ☐ **A** Turn early, when a trail is about 30 cm away
- ☐ **B** Keep to the outside of the mat
- ☐ **C** Head for the middle, where there is most space at the start
- ☐ **D** Wait until a trail is 15 cm away before turning
- ☐ **E** Never stand still

> **Answer:** A, B, E. The middle fills up first, and a robot that turns at 15 cm needs space it may not have.

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

```
import math
trail = [(0, 30), (10, 30), (20, 30)]
x, y = 8, 12
closest = min(math.hypot(px - x, py - y) for px, py in trail)
print(round(closest, 1))
```

Answer:

```
18.1
```

The point (10, 30) is 2 across and 18 up from (8, 12), about 18.1 cm. The other two points are farther.

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

```
found = None
print("line ahead?", bool(found))
found = [200, 5]
print("line ahead?", bool(found))
```

Answer:

```
line ahead? False
line ahead? True
```

`None` counts as false and a list with something in it counts as true, which is how `if found:` works.

6. Does `info()['trails']` include your own trail? [1 mark]

- ☐ **A** Yes, and touching your own trail puts you out too
- ☐ **B** No, only the other robots' trails
- ☐ **C** Only the last 10 cm of it
- ☐ **D** Only in team games

Answer: A. Every robot leaves a trail, yours included, so the mat gets smaller for you as well.

The task: thread the gap

Two trails lie across the mat, each with a gap at one end, and the gaps are at opposite ends. Drive from the start to the green zone at the top without touching a trail. In this task the trails are low walls, 6 cm high, so the depth sensor sees them and the line detector does not.

```
from bugbot import *
connect()

while position()[1] < 85:
    forward(45)
    wait(0.1)

stop()
```

The hint students can ask for: Each trail has a gap at one end, and the gaps are at opposite ends. Cross the mat between the trails, not over one.

A solution

```
from bugbot import *
connect()
def across(to_x):
    while abs(position()[0] - to_x) > 2:
        if position()[0] < to_x:
            right(45)
        else:
            left(45)
        wait(0.1)
    stop()
def up_to(y):
    while position()[1] < y:
        forward(45)
        wait(0.1)
    stop()
across(30)
up_to(41)
across(-30)
up_to(72)
across(20)
up_to(84)
stop()
```

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