WORKSHEET

10.4 Trails and tight spots

BugBotLab

Competing · Robot club · about 15 min

Name _____ Class _____ Date _____

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

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)`

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

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))
```

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

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

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

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()
```

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

Do it on the robot

www.bugbotlab.com/learn/10-4-trails-and-tight-spots/

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

Challenges

1. Use `distance()` or `scan()` to spot the trail ahead and steer for the gap rather than driving a fixed path.
2. Do it again with `drive()` only, never stopping.
3. Count how many times you changed direction. Fewer is usually faster.