

Turtle Mazes — teacher answer key

These are the solutions the group goes through on the “Check Together” slides. One line per card as laid. Nested (indented) lines are a frame/def body. Diamond detours are optional — listed per level; the card sequence is the no-detour shortest route. Verified by `validate.py` (BFS shortest = these counts).

Part 1 Levels 1–4

Level 1 — 01-s-bend

3×5 • START (0,0) facing S • 8 forwards, 2 turns • 0 one-ways

Card counts: flat cards 10 (= 8 forwards + 2 turns)

```
move_forward()
move_forward()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
```

⇒ first `for 4: [forward]` taste (4 straight on leg 1)

Level 2 — 02-zigzag

5×5 • START (0,0) facing S • 12 forwards, 4 turns • 0 one-ways

Card counts: flat cards 16 (= 12 forwards + 4 turns)

```
move_forward()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
```

⇒ loops fit but don't pay yet; detour +2 fwd +3 turns

⇒ diamond (1,2) spur off (2,2): on arrival facing N:
`left, forward, around, forward, left`

Level 3 — 03-weave

6×5 • START (0,0) facing S • 15 forwards, 6 turns • 0 one-ways

**Card counts: flat cards 21 (= 15 forwards + 6 turns); with
for N cards: 19 cards**

```
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
move_forward()
move_forward()
```

⇒ two 3-runs: `for 3: [forward]` is genuinely cheaper; detour +4 fwd +2 turns

⇒ diamond (2,0) spur off (2,2): on arrival facing E:
`left, forward ×2, around, forward ×2`

Level 4 — 04-spiral

6×6 • START (0,0) facing E • 23 forwards, 6 turns • 0 one-ways

**Card counts: flat cards 29 (= 23 forwards + 6 turns); with
for N cards: 18 cards**

```
move_forward()
move_forward()
move_forward()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
move_forward()
turn_right_90()
move_forward()
```

```

move_forward()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()

```

```

move_forward()
turn_right_90()
move_forward()
turn_right_90()
move_forward()

```

⇒ runs 5/5/4/4/3: five loop cards stand in for 21 of the 23 forwards; looped variant: for 5: [forward] , right , for 5: [forward] , right , for 4: [forward] , right , for 4: [forward] , right , for 3: [forward] , right , forward , right , forward

⇒ diamond (0,3) spur off (1,3): on arrival facing N: left, forward, around, forward, left

Part 2 Levels 5–7

Level 5 — 05-open-maze

6×6 • START (0,0) facing S • 10 forwards, 4 turns • 0 one-ways

Card counts: flat cards 14 (= 10 forwards + 4 turns)

```

move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_right_90()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()
turn_right_90()
move_forward()

```

⇒ two 3-runs + one 2-run (for 3 twice); detour +4 fwd +3 turns

⇒ diamond (3,1) spur off (2,2): on arrival facing E: forward, left, forward, around, forward, right, forward (back facing W; the continuation south needs one more turn)

⇒ dead-end trap: (4,4)→(4,3)→(5,3); escape with around, forward

Level 6 — 06-one-way-maze

5×5 • START (0,0) facing S • 14 forwards, 5 turns • 4 one-ways

Card counts: flat cards 19 (= 14 forwards + 5 turns)

```

move_forward()
move_forward()
move_forward()
turn_left_90()
move_forward()
turn_left_90()
move_forward()

```

```

move_forward()
move_forward()
turn_right_90()
move_forward()
turn_right_90()
move_forward()
move_forward()
move_forward()
move_forward()
turn_left_90()
move_forward()
move_forward()

```

⇒ straights 3/3/4 exist, but the one-ways are the axis

⇒ one-ways: route legs (0,1)→(0,2) and (1,2)→(1,1) are forward-only; loop corridor enters only via (3,1)→(2,1) and (2,2)→(3,2), so it is never a shortcut

Level 7 — 07-long-maze

6×6 • START (0,0) facing W • 30 forwards, 10 turns • 4 one-ways

Card counts: flat cards 40 (= 30 forwards + 10 turns); with def jog + 5 calls: 26 cards

```

turn_around()
def jog():
    move_forward()
    move_forward()
    move_forward()
    move_forward()
    jog()
turn_right_90()
move_forward()
turn_right_90()
jog()
turn_left_90()

```

```

move_forward()
turn_left_90()
jog()
turn_right_90()
move_forward()
turn_right_90()
jog()
turn_left_90()
move_forward()

```

```

turn_left_90()
jog()
turn_right_90()
move_forward()

```

⇒ five identical 6-cell crossings: `def jog(): forward × 5`, call it — the stretch task's referent

⇒ one-ways (1,2)→(2,2), (4,3)→(3,3), (0,3)→(0,4), (5,4)→(5,5) sit mid-run on route straights: overshoot/retreat is illegal

⇒ dead-end trap: (3,4)→(3,5) via `right, forward`; escape `around, forward, right`

Part 3 Caveats — read before signing off

1. **L5 is deliberately shorter than L4 (10 forwards vs. 23).** L4's locked spiral is 23 forwards; L5's locked source maze (SVG 05, arrows removed) is 10 — and removing arrows cannot shorten or lengthen a route. The break in the length ladder is inherited from the source set, recorded in the spec, not fudged: the how_to doc calls maze 05 “the hardest in the set not because its solution is longest (10 moves, versus 23 for maze 04's spiral)”. Difficulty ≠ route length; treat the reaction at L5 as the lesson.
2. **L7's `turn_around()` is the start-facing fix, not a stray card.** On a cell grid where turns cost no cells, a *shortest* route never revisits a cell — a mid-route 180° is provably removable from any “shortest” solution. The only way a genuine `turn_around()` belongs in a shortest solution is a start facing that opposes the first corridor: L7 starts (0,0) facing W while the first corridor runs E, so it opens `turn_around()`. A mid-route fold-back in L7 would be a spec bug, not a student error.

L7 count check: 26 cards on the table — 1 `def` card + 5 body cards in its frame + 20 play cards — replaying as 30 forwards, 10 turns (1 × `turn_around()` + 5 × `right` + 4 × `left`) — the five `jog()` calls each re-run the 5-card body.