

# A-maze-ing Robots mBot Challenge



Team Name: \_\_\_\_\_

Team Members: \_\_\_\_\_

Robot Name: \_\_\_\_\_

## Direction Finding Strategy

### Hints:

How will your robot know where it is?

Avoid the walls?

Follow the lines?

Both? Something else?

Watch out for dead ends!

## Maze Solving Strategy

### Hints:

How will your robot decide where to go?

Follow the left/right wall?

Always turn left/right?

Flip a coin and choose the direction?

Can it reverse?

# A-maze-ing Robots mBot

## Hints & Tips



### Maze Solving

- Some different strategies for solving mazes
  - Try to keep wall always on your left/right side
  - Always turn left/right when you have to choose
  - Randomly pick a direction to turn when you have to choose
- Watch out for dead ends!
  - What will your bot do if it hits a dead end? There are some in the maze, make sure your robot knows how to reverse or it might get stuck!
  - Puzzle: How will you know when to stop reversing?
- Pretend you're a robot!
  - Get your team members to test their maze strategy with one of you acting as the robot, and only move like your robot would, only using the instructions you've planned – did you get stuck? Could you get yourself out?

### Working with mBot

- The sprites in the stage are available to you, and if you put values into variables, you can make the sprite “say” those values, or just display those variables on screen. This might help with debugging your code!
- mBot can play notes – why not have it play a different note for each direction of turn, when it detects an obstacle, when it gets stuck, etc. while you are testing your code, to help you debug on the fly
- mBot can move very quickly or very slowly, but I recommend starting at around 30% speed and working your way up. Otherwise mBot might be moving too quickly to react!
- Don't forget to code a “when press <some key>, stop” type of block, so you can stop your bot running away when you need to!

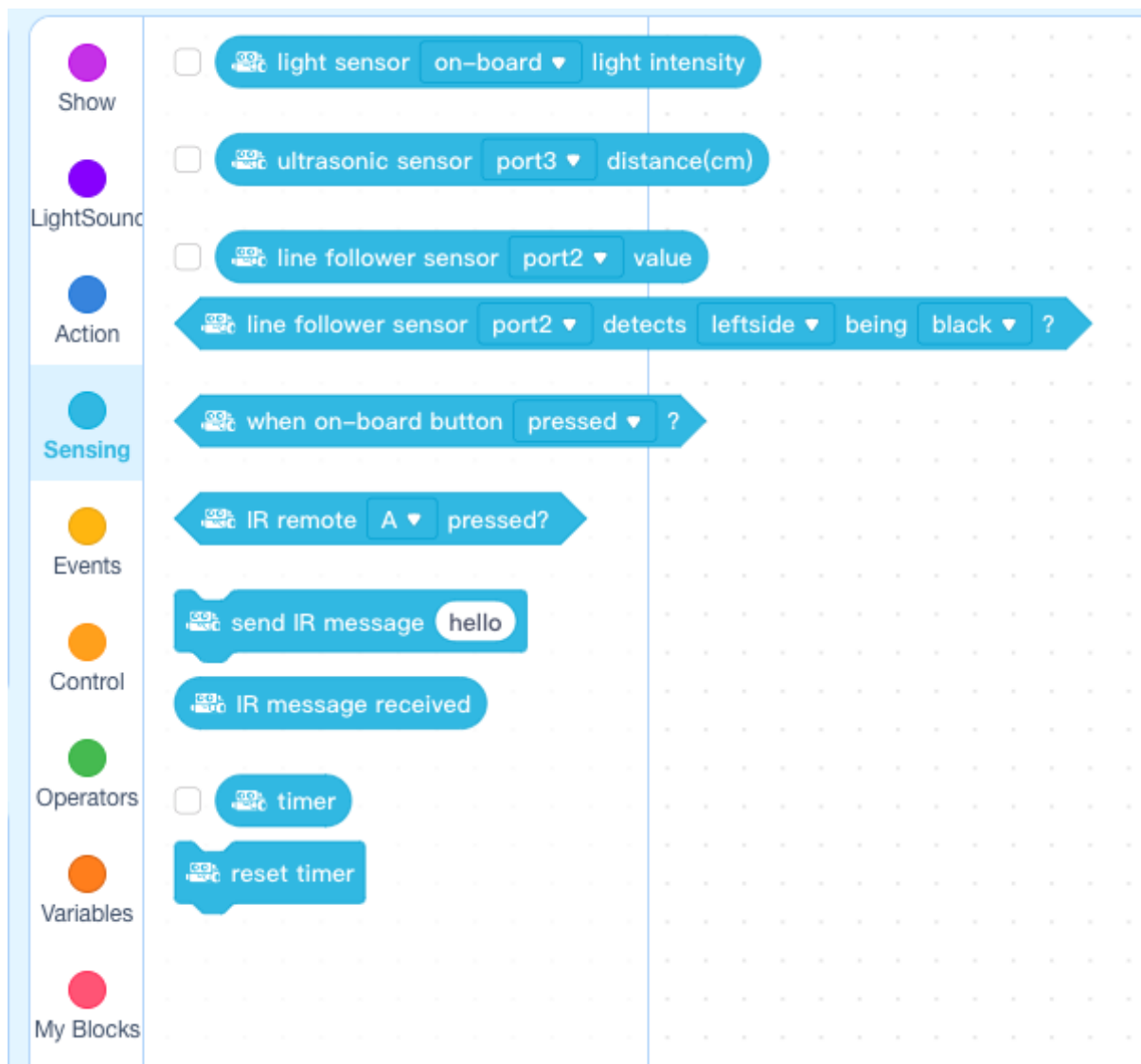
# A-maze-ing Robots mBot

## Hints & Tips



### mBot Sensing Blocks

This is where the line follower and ultrasonic sensor blocks can be found, these will be valuable tools for your robot once it's in the maze!



The screenshot shows the mBot programming interface with the 'Sensing' category selected in the left sidebar. The main workspace displays several sensing blocks:

- light sensor on-board light intensity
- ultrasonic sensor port3 distance(cm)
- line follower sensor port2 value
- line follower sensor port2 detects leftside being black ?
- when on-board button pressed ?
- IR remote A pressed?
- send IR message hello
- IR message received
- timer
- reset timer

# A-maze-ing Robots mBot Hints & Tips



If there's a black line straight ahead, keep following it forward!

```
if line follower sensor port2 detects all being black ? then
  move forward at power 30 %
```

If there's something less than 2cm ahead of you, stop, and turn right til you have more than 2cm room in front of you

```
if ultrasonic sensor port3 distance(cm) < 2 then
  stop moving
  repeat until ultrasonic sensor port3 distance(cm) > 2
  turn right at power 50 %
```

mBot, flip a coin!

```
if pick random 0 to 1 = 1 then
  turn right at power 50 %
else
  turn left at power 50 %
```