



U8.1 A map of cells

Name _____

Class _____

Date _____

What this lesson is about

What a map has to do for a robot, why a grid is the usual answer, and what a cell costs.

- What a map is for
- Two families
- Resolution
- Indexing
- The pose is assumed known here

Questions

6 marks in all

1. What does this program print?

[1 mark]

```
CELL = 5.0
x, y = 102.4, 187.6
col = int(x / CELL)
row = int(y / CELL)
print(col, row)
print(col * CELL + CELL / 2, row * CELL + CELL / 2)
```

2. What does this program print?

[1 mark]

```
CELL = 5.0
grid = [0] * 40
x = -7.0
grid[int(x / CELL)] = 1
print(grid.index(1))
```

3. Why do occupancy grids beat feature maps for planning, despite using far more memory?

[1 mark]

- ☐ A They represent free space explicitly, and "nothing is there" is what a planner needs most
- ☐ B They store each feature with a covariance
- ☐ C They need fewer sensor readings to become confident
- ☐ D They suit a Kalman filter better

4. A 200 by 200 cm mat is mapped with 1 cm cells, each stored as a 4 byte float. How many bytes does the grid need? [1 mark]

5. Which cell size best follows the page's rule of thumb for a robot that must fit through 20 cm gaps, with a sensor whose noise is about 2 cm? [1 mark]
- ☐ A 5 cm
 - ☐ B 1 cm
 - ☐ C 20 cm
 - ☐ D 40 cm
6. Every mapping task in U8 gives the robot its true pose from the overhead camera. Why? [1 mark]
- ☐ A Mapping with known poses is a tidy, solved problem, whereas building a map while localising against it lets the two errors feed each other
 - ☐ B The robot's sensors cannot measure distance without the camera
 - ☐ C Occupancy grids cannot be built from odometry
 - ☐ D The camera is more accurate than any map could ever be

The task: which cell did that reading land in?

The robot stands at (100, 50) on the mat facing straight up it. Print `cells:`, how many cells a 5 cm grid needs for this mat, and `hit col:` and `hit row:`, the cell the reading ahead lands in.

```
from bugbot import *  
connect()
```

```
CELL = 5.0  
W = 40
```

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



Do it on the robot

www.bugbotlab.com/learn/u8-1-a-map-of-cells/

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

Challenges

1. Work out how much memory a 1 cm grid of this mat needs if each cell is a 4 byte float.
2. Print the centre of the hit cell in centimetres. How far is it from the point the reading actually gave?
3. At what cell size would a 20 cm doorway be at risk of closing up?