

Homework 7

Submit Assignment

Due Apr 26 by 11:59pm **Points** 12 **Submitting** a file upload **File Types** py
Available Apr 16 at 8am - Apr 29 at 11:59pm 14 days

Your task this week is to guide our agent to a treasure hidden at the bottom of the ocean.

The task is presented as a game and the mechanics have been implemented for you. The starter code is available in [HW7.zip](#).

Once you download the code, you can play the game in discovery mode (without help) by typing:

```
python treasurehunt.py 6 discovery
```

To play the game on a bigger grid, you can try:

```
python treasurehunt.py 10 discovery
```

Click on the 'SONAR' button to start taking sonar measurements. When you are ready to make a decision and dive, click on the 'DIVE' button.

Your task is to help the player/agent find the treasure more reliably by implementing the following:

1. The *update* method in the belief.py module. The Belief class is used to track the belief distribution (where we think the treasure is) based on the sensing evidence we have so far. The distribution is initialized to a uniform distribution in `__init__`. Your task is to update it with each evidence. Don't forget to normalize!

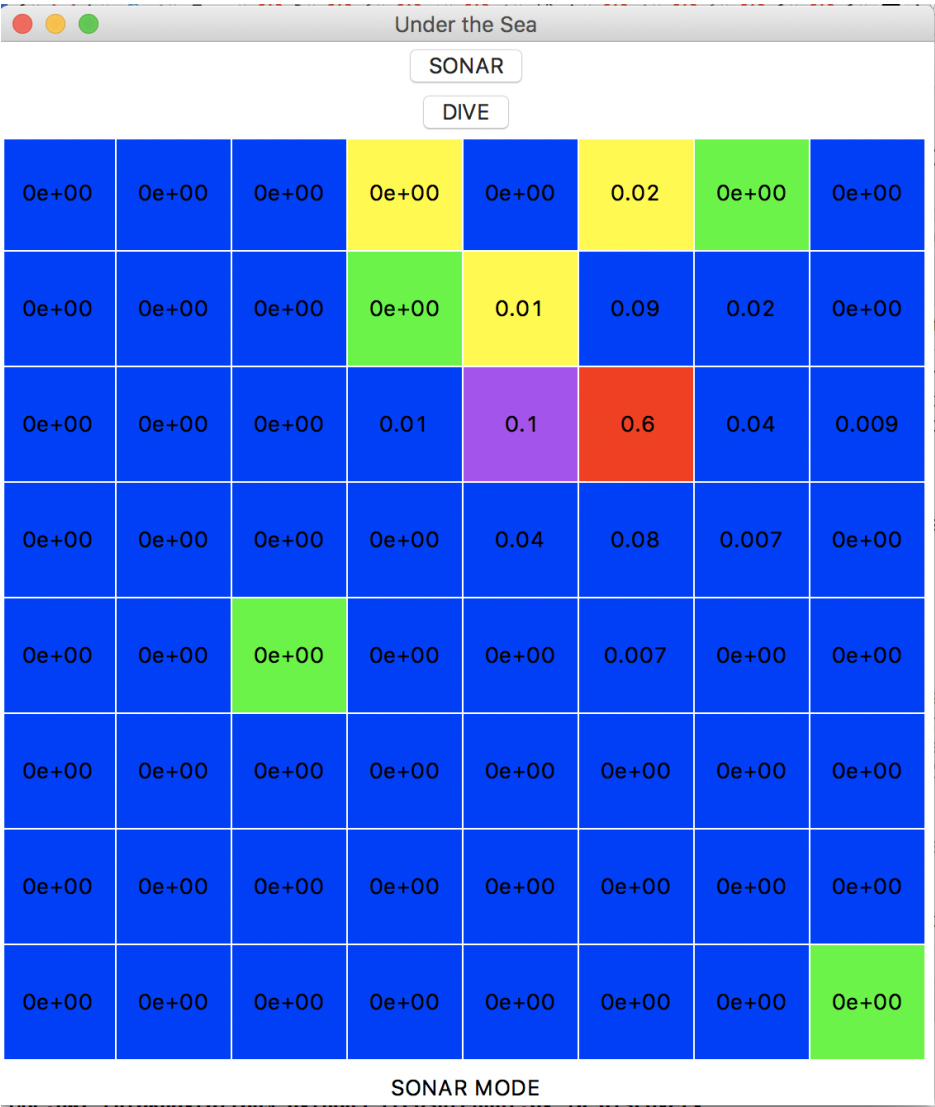
To see the distribution on the grid, start the game in 'guided' (default) mode:

```
python treasurehunt.py 8 guided
```

2. The *recommend_sensing* method in the belief.py module. The method should return the position where we should take the next sonar measurement in the grid. The position should be the most promising unobserved location. If all remaining unobserved locations have a probability of 0, the method should return the unobserved location that is closest to the (observed) location with the highest probability. The recommended location is displayed in purple on the grid - as shown in the screenshot below.

You only need to modify and upload beliefs.py.

Please make sure yo read the grading rubric to ensure full credit.



1.

Homework 7 Grading Rubric

Criteria	Ratings		Pts
The update method: distribution is updated based on the model	3.0 pts Full Marks	0.0 pts	3.0 pts
The update method: normalization <i>The probabilities for all locations in the grid must add up to 1.</i>	2.0 pts Full Marks	0.0 pts	2.0 pts
The update method: the open set (unobserved locations) is updated correctly	1.0 pts Full Marks	0.0 pts	1.0 pts
The recommend_sensing method: most promising unobserved location <i>An open (unobserved) location with the highest probability is returned</i>	2.0 pts Full Marks	0.0 pts	2.0 pts
The recommend_sensing method: if all remaining unobserved locations have a probability of 0, return the unobserved location that is closest to the (observed) location with the highest probability.	2.0 pts Full Marks	0.0 pts	2.0 pts
The manhattan_distance function in utils.py is used	1.0 pts Full Marks	0.0 pts No Marks	1.0 pts
The closest_point function in utils.py is used	1.0 pts Full Marks	0.0 pts No Marks	1.0 pts
Total Points: 12.0			