Exercise 1: Localization

Consider the world shown below. A robot moves counter-clockwise (in a deterministic way) in a circular corridor containing 10 grid cells. In some grid cells are landmarks installed. If the robot is in a cell with a landmark it will detect it with a probability of 80%. If there is no landmark within the grid cell, the robot’s sensors will detect one with a probability of 40%. Compute for each grid cell the probability that the robot is in a particular cell after the following sequence of movements and measurements:

1. The robot detects a landmark.
2. The robot moves 3 grid cells forward.
3. The robot detects again a landmark.
4. The robot moves 4 grid cells forward.
5. The robot detects no landmark.
Exercise 2: Implementation of a simple discrete filter:
Implement a simple discrete filter for the solution of exercises like the one above. Please use the Java template and data files linked on our web page. Input file 1 matches the situation in Exercise 1 and can be used as a test case for your implementation.

(a) Implement the algorithm with a deterministic motion model and the sensor model of Exercise 1. Add a method that interprets the final belief distribution after all inputs have been processed.

(b) Exchange the deterministic motion model for an undeterministic one that approximates a Gaussian distribution:

$$P(x_t = c + a - 1|x_{t-1} = c, a) = \frac{1}{4},$$
$$P(x_t = c + a|x_{t-1} = c, a) = \frac{1}{2},$$
$$P(x_t = c + a + 1|x_{t-1} = c, a) = \frac{1}{4}.$$  

($x_t$: cell index at time $t$, $c$: cell index, $a$: action (number of cells moved)).

(c) Describe shortly, how you would implement the filter as a particle filter and what the advantages and disadvantages would be.

Please send your implementation and output files to plagem@informatik.uni-freiburg.de.