Name: ____________________________ ITU ID: ______________________ Signature: ____________

İSTANBUL TECHNICAL UNIVERSITY

Department of Computer Engineering

BLG456E – Robotics – Spring 2013

Final exam.

BLG456E FINAL

Duration: 120 minutes

There are 10 questions.

Rules:
- Not open-book. No extra notes or papers are allowed.
- Cellphones must be put away. Basic calculators are allowed.
- Answers must be in English.
- **Show your working.**
- Put at least your name or ID on all pages.
- If you write in the margins (you should not need to), indicate under the relevant question.

Question 1 (5 pts): What is an observation model? Give one way that an observation model can be acquired.

A mathematical/computational/probabilistic model of how world/actual states & events can generate sensory states/events/observations. Can be specified through prior knowledge or learnt automatically.

A robot is trying to localise in a 4x4 grid, which is surrounded by a wall. It currently has ***no prior information*** *about its current location. It can sense whether it is close to a wall or if there is no wall. Its wall sensor is active with the following probabilities conditional only on whether the current state is next to a wall:*

```
P( WallSensor=on | NextTo(state)=Nothing ) = 0.8    P( WallSensor=on | NextTo(state)=Wall ) = 0.6
P( WallSensor=off| NextTo(state)=Nothing ) = 0.2    P( WallSensor=off| NextTo(state)=Wall ) = 0.4
```

Question 2 (15 pts): Fill in the location *likelihoods* in the grid below left if a wall is detected (`WallSensor=on`). Fill in location *probabilities* in the grid on on the bottom right. You can use fractions instead of decimals if you wish.

0.6	0.6	0.6	0.6
0.6	0.8	0.8	0.6
0.6	0.8	0.8	0.6
0.6	0.6	0.6	0.6

.57	.57	.57	.57
.57	.77	.77	.57
.57	.77	.77	.57
.57	.57	.57	.57

Name: ___________________________ ITU ID: _____________________ Signature: ____________

An iterative forward search motion planning algorithm uses the following 3 motion primitives for a car with no reverse gear:

Motion primitive 1:

- Move directly forward at $10\,ms^{-1}$ for $\frac{1}{10}s$ ($\dot{x}=10\,ms^{-1}$, $\dot{\theta}=0\,rad\ s^{-1}$, $\Delta t=\frac{1}{10}s$)

Motion primitive 2:

- Move arcing left at $10\,ms^{-1}$ for $\frac{1}{10}s$ ($\dot{x}=10\,ms^{-1}$, $\dot{\theta}=\frac{\pi}{4}rad\ s^{-1}$, $\Delta t=\frac{1}{10}s$).

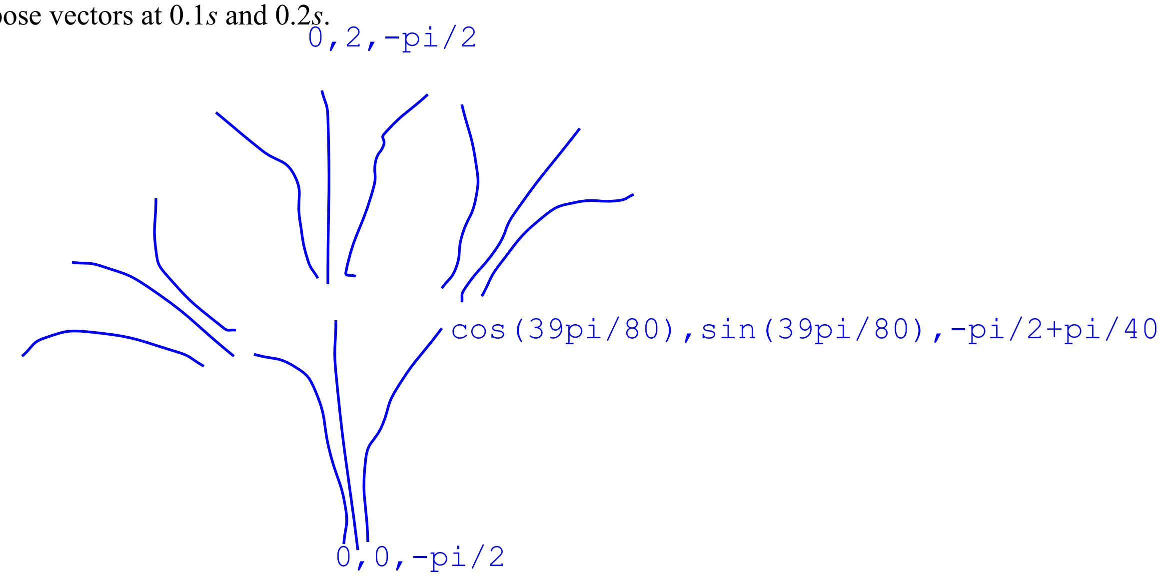
Motion primitive 3:

- Move arcing right at $10\,ms^{-1}$ for $\frac{1}{10}s$ ($\dot{x}=10\,ms^{-1}$, $\dot{\theta}=-\frac{\pi}{4}rad\ s^{-1}$, $\Delta t=\frac{1}{10}s$).

Question 3 (5 pts): How many possible paths are there after 1 second? (you can write an expression rather than a number).

3 to the power of 10

Question 4 (15 pts): Draw the possible paths that the iterative forward search algorithm would produce after 0.2*s* if the initial pose is $x=0, y=0, \theta=-\frac{\pi}{2}$. Assume that the direction $\theta=0$ is along the positive x axis and that rotations are expressed counter-clockwise (anti-clockwise). Annotate the diagram with *all* the possible robot pose vectors at 0.1*s* and 0.2*s*.



Question 5 (10 pts): Name a different path planning algorithm and compare the iterative forward search algorithm to it in terms of *efficiency* and *completeness*. Give reasoning.

RRT - More efficient because designed to cover space efficiently
- RRT is probabilistically complete while iterative forward search is resolution complete

Grid a* - Generally more efficient depending on resolution of grid. Also re-usable decompositions and plans.
- a* is complete if the grid is exact, otherwise resolution complete. iterative forward search also resolution complete

etc.

Question 6 (15 pts): A robot arm is supposed to follow a trajectory. A *P-controller* is installed to control the joints of the arm. Unfortunately, the arm never quite stays on the target and instead wiggles (oscillates) around target. Referring to how a P-controller works:

- Describe what could be going wrong.
- Suggest 2 possible solutions to this problem and their possible drawbacks, if any.

Gain is too high causing oscillations. OR: the feedback is too strong meaning that the controller overshoots and must continually compensate.

Sol 1: Decrease gain: Problem: Can slow convergence.

Sol 2: PD control to reduce overshoot. Problem: Can slow convergence (but not as much as sol 1).

Question 7 (5 pts): In the area of robot audition, what is the "cocktail party effect"?

The robot must extract/distinguish between waveforms/sounds from different sources in a real 3D environment.

Question 8 (5 pts): You want a humanoid to learn to balance on one leg. Suggest a data structure and algorithm for learning to do this task.

P controller, evolutionary algorithm.
Mixture of Gaussians controller, EM.
etc.

Question 9 (10 pts): Give two examples each of passive and active sensing. Give one example each of passive and active localisation.

Passive sensing: Camera, battery monitor, light sensor, microphone...
Active sensing: Laser scanner, radar, time-of-flight camera...
Passive localisation: Maximum likelihood, radar-based, particle filter...
Active localisation: Acting to sense, moving head to look, information gain...

Question 10 (15 pts): A robot is rotated $\pi/2$ radians and then translated (5,-7) where the unit of distance is metres.

1) Give the matrix representing this transform.
2) If the robot's wheel is located at (2,2) (in the original frame of reference) before the transform, what is the position of the robot's wheel after the transform?

$$\begin{bmatrix} \cos(pi/2) & -\sin(pi/2) & 5 \\ \sin(pi/2) & \cos(pi/2) & 7 \\ 0 & 0 & pi/2 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 5 \\ 1 & 0 & 7 \\ 0 & 0 & pi/2 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 5 \\ 1 & 0 & 7 \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 9 \end{bmatrix}$$

Name: ___________________________ ITU ID: _____________________ Signature: ____________

Extra space for answers/working

- If you write answers here, indicate as such under the appropriate question.