Last updated on Tuesday, March 26, 2013

Assignment (2013 INB860)

Key points and due dates

- Demonstrations to the to tutor have to be performed during the scheduled lab sessions.
- Demonstrations start in **Week 10** and continue until Week 13
- Submit report by the End of **Week 13**
- Use **Blackboard** to submit your work
- Group size (from 1 to 3)
- Sent questions to *f.maire@qut.edu.au*
- Do not underestimate the workload. Start early. You are strongly encouraged to ask questions during the practical sessions. But, the closer the submission date, the less help/hints you will get...
- A criteria sheet (more detailed marking guide will be released later)

Overview

In this assignment, you will design and implement a number of behaviors on the Lego robot to solve a number of challenges.

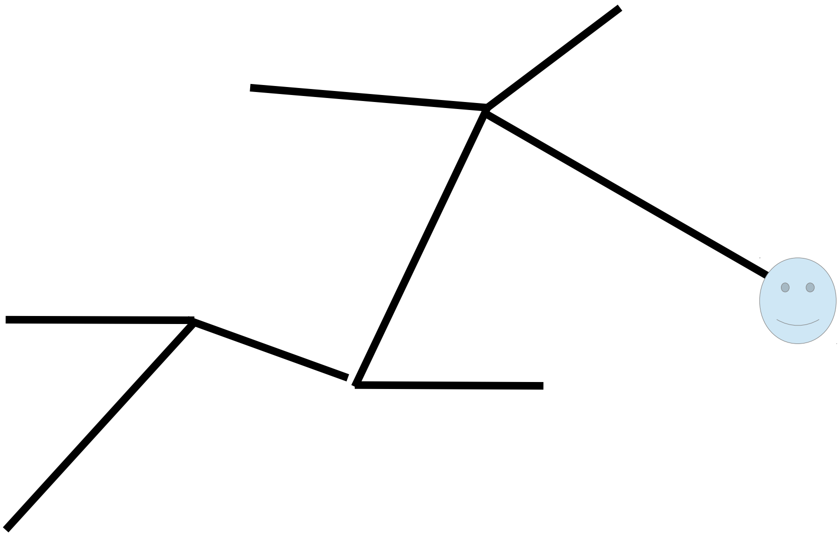
You will demonstrate your programs to the tutor during your lab sessions starting in Week 10 until Week 13. The tutor will record whether or not you have solved satisfactorily the different challenges in a criteria sheet. Detailed demonstration schedule at the end of this document.

The robotic arenas will be laminated A0 size posters with dark tape lines on a white background. Some cans and gray patches will also be used. You are only allowed to use the set of sensors contained in a single kit.

All *final messages* should be displayed for 10 seconds on the brick.

The angle between two consecutive branches of an intersection is constrained to be between 30 degrees and 150=180-30 degrees. This means that you can detect that you are passing an intersection by monitoring a virtual compass. The minimum length of a straight segment is 15cm.

Challenge 1 - Monkey business



The environment is a connected network **without cycles** of dark straight segments, and a can or mug (illustrated with a smiley in the diagram).

Behavior 1

Move along a straight dark line until you detect a can or the segment ends (here the tree is reduced to a single twig!). When a stopping condition is reached, stop the robot, beep and display the final message “can detected” or “segment end” according to the stopping condition.

Behavior 2

Move along a straight dark line until you reach an intersection or the segment ends. When a stopping condition is reached, beep and display the message “number of branches = n” where n is the number of branches of the intersection. If the robot has reached a leaf (that is, the segment ends), you should display “number of branches = 1” as the node leaf has one branch.

Behavior 3

Explore the network driving along the segments. The tutor will choose the start position. The start position can be on any segment. However, the start position will be close to a node, but facing away from the node. This way, the robot should detect more reliably a change of direction when it reaches the next node.

While exploring the network you should display at each intersection the number of branches on one line of the brick, and on another line you should display a label number for the current intersection (you are naming the intersections with numbers). If this is the first time you visit an intersection you should display “New node n” when n is the node label. If the node is already known (visited), you should display “Known node n” where n is the node label that you assigned to the node at the first visit. You should also display the current compass direction on a separate line. Stop when you have visited all the nodes. Display a final message “n nodes found” where n is the number of nodes in the tree (including leaves).

Behavior 4

Explore the network driving along the segments like in Behavior 3. When you find a can (cans are only located at leaf nodes), beep, pause the motors and display for 5 seconds the message “can detected at node n” where n is the label of the leaf node. After the 5 seconds, resume the exploration until you have visited all the nodes. At this point, pause the motors and and display for 5 seconds the message “n nodes found” where n is the number of node in the tree (including leaves). Then return to the leaf where the can was detected, and display a final message “can was here”.

IMPORTANT NOTES

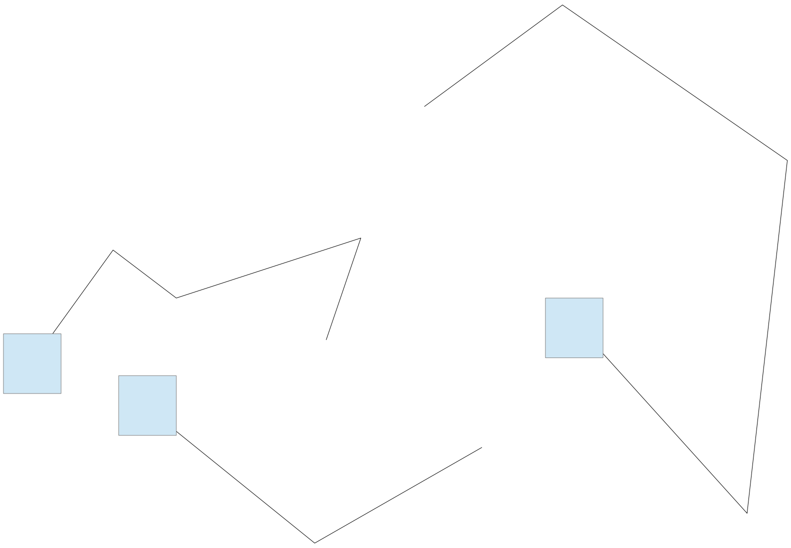
The tutor will remove the can after its initial detection. It means that you have to record in a map where the can was. You should use the shortest path to go from the node that was discovered last in the exploration phase to the node where the can was detected.

Behavior 5

Behavior 5 is an extension of Behavior 3. Create a text file to represent the tree incidence matrix with the following format; each line corresponds to a node. First number is the node label, the remaining numbers on the line are the labels of its neighbors.

Challenge 2 – String archipelago

.



The environment is an archipelago of string islands (set of polygonal lines). One end will have a grey patch (the hotel), the other end (the jetty) will be free of grey patch. If you follow the direction of the jetty, you will hit the next island. The group of islands form a cycle. There will be at most 8 islands in the archipelago.

Behavior 1

Move along the shore of an island until you detect the grey patch. When you detect the grey patch, you should beep, stop and display the final message “at the hotel”.

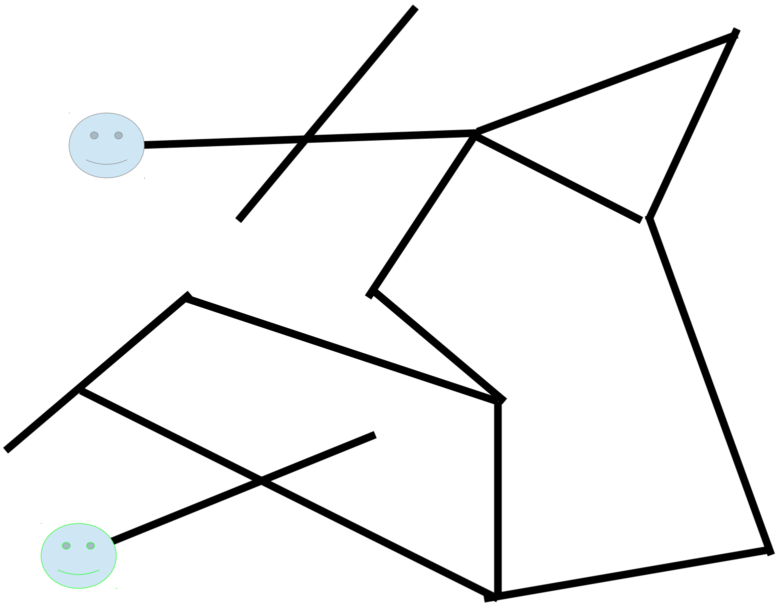
Behavior 2

Behavior 2 is a continuation of Behavior 1; instead of completely stopping when you find the hotel, make a U-turn, go to the end of the jetty, then stop and display the final message “At the jetty end”.

Behavior 3

Explore all the islands, starting on the shore of *island 1*. While exploring you should display the island label you are on. When the robot starts, it is on *island one*, the next island it encounters will be called *island 2*. You increment the island number each time you reach a new island. However, when you detect that you are back on a previously visited island, you should stop, beep and display the final message “back on island n” where n is the label you gave the island the first time you visited it.   IMPORTANT NOTES the tutor will have the islands on separate small posters, and will be able to move the islands before you revisit them. Therefore, you cannot rely on dead-reckoning to decide whether you are back on a previously visited island. You need to compare the shape of the islands.

Challenge 3 – Web crawler



The environment is a network (**with cycles**) of dark segments, and some cans and grey patches. In a nutshell, this is the same as Challenge 1 but cycles in the network are now allowed.

Behavior 1

Explore the network driving along the segments. The tutor will choose the start position (as for Challenge 1). While exploring the network you should display at each intersection the number of branches on one line of the brick, and on another line you should display a label number for the current intersection (you are naming the intersections with numbers). If this is the first time you visit an intersection you should display “New node n” when n is the node label. If the node is already known (visited), you should display “Known node n” where n is the node label that you assigned to the node at the first visit. You should also display the current compass direction on a separate line. Stop when you have visited all the nodes. Display a final message “n nodes found” where n is the number of nodes in the graph (including leaves).

Behavior 2

Explore the network driving along the segments like in Behavior 1. When you find a can (cans are only located at leaf nodes), beep, pause the motors and display for 5 seconds the message “can detected at node n” where n is the label of the leaf node. After the 5 seconds, resume the exploration until you have visited all the nodes. At this point, pause the motors and and display for 5 seconds the message “n nodes found” where n is the number of node in the graph (including leaves). Then return to the leaf where the can was detected, and display a final message “can was here”. Like in Challenge 1, the tutor will remove the can after its initial detection. It means that you have to record in a map where the can was. You should use the shortest path to go from the node that was discovered last in the exploration phase to the node where the can was detected.

Behavior 3

Behavior 3 is an extension of Behavior 1. Create a text file to represent the incidence matrix of the graph with the same format as for Challenge 1; each line corresponds to a node. First number is the node label, the remaining numbers on the line are the labels of its neighbors.

Behavior 4

Behavior 4 is a variation of Behavior 2. This time, there will be a can and a grey patch at two different leaves. Explore the network until you find both the can and the grey patch. As before, you should display information about the nodes traversed and the objects detected. Once you have detected the can and the grey patch you should go from one object to the other using the shortest path in the subgraph that you have explored so far. In other words, you should not wait until you have fully explored the entire graph to make the trip between the two objects.

Deliverables

On top of demonstrating the robot behaviors to the tutor, you should submit via Blackboard a zip file containing

1. A README.TXT file containing a table of contents of the zip file you are submitting
2. A report limited to 8 pages of text (be concise!) in pdf format
   - explaining concisely your approach for each behavior
   - describing the performance and limitations of the behaviors you have implemented
3. A statement of completeness (list which behaviors you have attempted, whether they meet the specifications and whether there are any limitations)
4. All c files with proper commenting

Draft Marking Scheme

- Report: 8 marks
- Code quality (readability, simplicity, structure, genericity, in line documentation): 8 marks

Demonstration Schedule

Week Number	Demonstrate	Marks
10	C1B1, C2B1, C2B2	2+2+2
11	C1B2, C2B3	3+4
12	C1B3, C1B4, C1B5, C3B1	5+5+3+5
13	C3B2, C3B3,C3B4	5+3+5

CxBy means Challenge x Behavior y

Have fun while learning!