

Week 03, Lab 03

Weight: 1%

Due: End of your stream's week 3 lab session (via `sync`)**Pre-lab Preparation:**

- Week 1, 2 Lectures, Week 1, 2 Lab
- Lecture: Wk03 Day 007, 008

Lab Activities:

Remember to `sync` to obtain the lab starting code.

Exercise 1: Robot Sequence

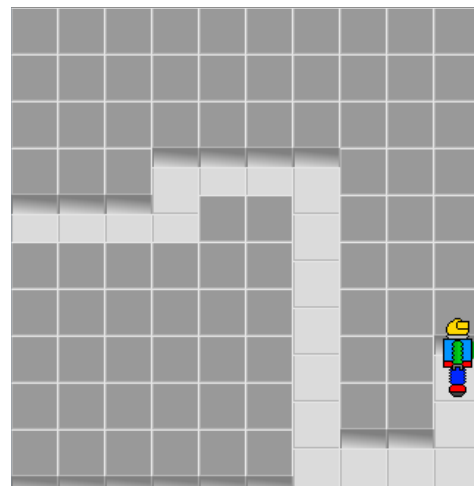
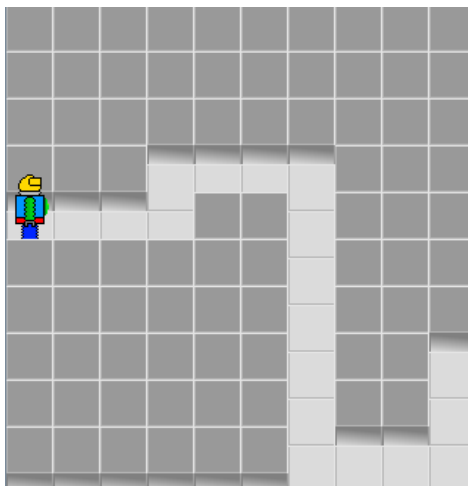
Navigate to the directory: `~/p1.2015s1/lab03/working_copy/ex01/`

Open the `lab03ex01.c` in the editor. Replace the number `1234567` with your ID number. Be careful to leave the return part and the semicolon in place. If your ID number starts with a zero, leave that off.

Save the file, close the editor or switch to another virtual terminal, and run: `make test`

You will see a Robot World like that shown in the figure below left. Your task is to write code in the function `move_along_corridor()` to move the robot from one end of the corridor to the other using sequences of Robot World function calls such as:

- `move_robot_forwards();`
- `turn_robot_left();`
- `pick_up_item_with_robot();`
- `drop_item_from_robot();`



The robot should finish at the eastern end of the corridor and facing east, as shown in the figure above right.

Open `lab03ex01.c` in the editor and begin! Once your code is complete, run `make test` again and ensure the unit tests pass.

Exercise 2: Planning

Navigate to the directory: `~/p1.2015s1/lab03/working_copy/ex03/`

Note that is **ex03**! Run **make test** a few times. You should see that each time the test runs, the corridor is a different shape.

Programs often need to deal with variation in their stating conditions.

Your next task is to write a version of **move_along_corridor()** using loops and if-statement that can move along any corridor.

First you should plan your algorithm using a flowchart. Log into a lab computer and start Microsoft Visio.

This is a tricky task, so feel free to confer with your colleagues to work it out. It will involve at least two loops, one to walk along a corridor of varying length, and one to walk along *each* corridor, and at least one test (if statement) to determine which way the next corridor goes.

Exercise 3: Random corridors

Now implement your algorithm from Exercise 2 in:

`~/p1.2015s1/lab03/working_copy/ex03/lab03ex03.c`

Run **make test** several times and ensure that your code compiles and passes all tests every time it is run.

Document any collaboration you undertook when making your flowchart in to code comment in **lab03ex02.c**.

Once complete, have a TA review your completed exercises 1, 2 and 3 for this lab session. See the end of this document for the review questions.

Exercise 4: Tidying beepers

Navigate to the directory: `~/p1.2015s1/lab03/working_copy/ex04/`

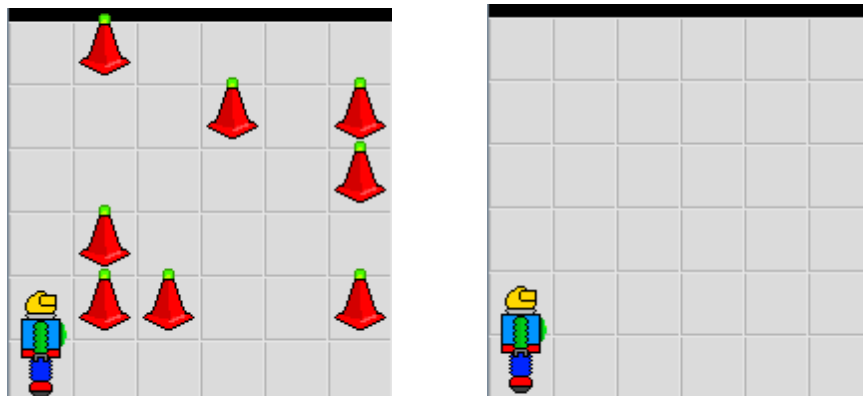
Run: `make test`

You will find that `make` does not succeed because a needed function has not been defined in `lab03ex04.c`.

Open that file and define an appropriate dummy function now. A dummy function has the correct signature but an empty body. E.g. this is a dummy function for a function called `do_something()`.

```
void do_something()
{
}
```

Once you have succeeded in defining that function you will see a robot world similar to that on the left below.



Again, the code for this question generates a different pattern of beepers based on your student ID number. Open the `lab03ex04.c` in the editor. Replace the number `1234567` with your ID number. Be careful to leave the return part and the semicolon in place. If your ID number starts with a zero, leave that off.

Run, `make test`, you will now see your pattern of beepers.

Write code for the needed function that causes the robot to pick up all of the beepers, then move to location (0, 0), facing east. So the final state of the world should be as shown on the right above. Be sure to use good programming style. You may use loops and methods to improve your code if you wish.

Run `make test` and check that all the unit tests pass.

Exercise 5: Student bedroom

The robot world in Exercise 5 is even messier than in Exercise 4, in the sense that the beepers will be arranged randomly each time make test is run.

Generalise your code from Exercise 4 to cope with this new scenario.

Week 03, Lab 03 Submission:

Run the **sync** command to submit your completed lab work.

Shutdown your Raspberry PI by pressing **ALT-CTRL-DEL**. Power-down and pack up your Raspberry Pi kit.

Marking Criteria:

Have you completed each of the following? Have you submitted your code from lab?

Marking Criteria:	Week 03 Lab 03 Weight 1%	Yes	No
Ex 1:	You have replaced 1234567 with your own ID?		
	All tests pass?		
Ex 2:	Flowchart prepared?		
Ex 3:	All tests pass on repeated runs?		
	Collaborations documented?		
Ex 4:	All tests pass?		
Ex 5:	All tests pass on repeated run?		
	Total:		

Next activity: Final Week 3 Lecture and Week 3 Homework.