University of Glasgow | School of Computing Science

Assessed Coursework

Course Name	**Robotics Foundations**			
Coursework Number	**1**			
Deadline	**Time:**	**2:30 pm**	**Date:**	**20 March 2025**
% Contribution to final course mark	**20**		**This exercise should take at most these many hours:**	**20 (max and indicative)**
Solo or Group ✓	**Solo**	✓	**Group**	✓
Submission Instructions	**Via Moodle**			
Who Will Mark This? ✓	**Lecturer ✓**	**Tutor**	**Other**	
Feedback Type? ✓	**Written ✓**	**Oral**	**Both**	
Individual or Generic?	**Generic**	**Individual ✓**	**Both**	
Other Feedback Notes				
Please Note: This Coursework cannot be Re-Done				

Code of Assessment Rules for Coursework Submission

Deadlines for the submission of coursework which is to be formally assessed will be published in course documentation, and work which is submitted later than the deadline will be subject to penalty as set out below. The primary grade and secondary band awarded for coursework which is submitted after the published deadline will be calculated as follows:

(i) in respect of work submitted not more than five working days after the deadline
   a. the work will be assessed in the usual way;
   b. the primary grade and secondary band so determined will then be reduced by two secondary bands for each working day (or part of a working day) the work was submitted late.

(ii) work submitted more than five working days after the deadline will be awarded Grade H.

Penalties for late submission of coursework will not be imposed if good cause is established for the late submission. You should submit documents supporting good cause via MyCampus.

The penalty for non-adherence to submission Instructions is 2 bands

Marking Criteria
See Page 4

Contents

Introduction

The coursework is based on what you have been working on in the labs. It should be developed using Python, ROS, and associated technologies, including Gazebo and RViz. Your solution should draw on the skills you have built up so far. The aim is to develop a solution to enable the FR3 robot to stack boxes. The tasks you must solve for this coursework are specified as follows:

- The FR3 must be able to pick and place boxes.
- Your ROS solution must locate spawned boxes and should not rely on hardcoded coordinates.
- The FR3 should be able to stack at least 4 cubes;
    - The robot must pick and place each box one at a time.

Tips: We suggest working out first how to pick and place boxes, then stacking them up and, finally, integrating all modules. You can use code from the labs.

NOTE: Remember to back up your work at all times!

This coursework should take you around 10 hours (average) to complete and covers the following Intended Learning Outcomes of the course[1]:

ILO 2. Formulate robot's information capabilities within robotic middleware and understand how data is transformed from basic control to robot actions

ILO 4. Apply techniques for path and motion planning that allows a robot to move

ILO 5. Design control/behaviour tasks for robot manipulators

ILO 6. Apply practical software engineering principles during the development of a robotic application

ILO 7. Understand the facilities provided by ROS and how to structure robot control software systems using ROS.

ILO 8. Be able to program a data processing pipeline consisting of a robot control system in the ROS environment to carry out a specific task, such as locating an object, grasping it and placing it in another location.

[1] https://www.gla.ac.uk/coursecatalogue/course/?code=COMPSCI4076

How can I effectively manage my coursework?

You are free to work on this coursework individually or as a group. Note that this coursework has been designed to be completed individually; however, we believe that peer learning is a useful technique for consolidating knowledge and solving a problem. You do not need to register your team; you will let us know who your team was during your submission (see the *What should be submitted* section).

If you are working as a group, you should always be involved in the solution. We will test your understanding of the whole coursework—**the coding part accounts for 2 marks only**.

If a teammate disappears, is unresponsive, does not contribute, etc., feel free to complete the coding aspect of the coursework. We will not mediate group problems; this is because each member will need **to answer 3 open-ended questions individually**, where we will assess your understanding and active involvement in the development of the solution. These 3 open-ended questions account for 18 marks. If you actively participate in developing the solution, you should be capable of answering all questions!

Materials

For this assessed exercise, you can either use your local WSL2/Ubuntu installation or the ConstructSim.

- For WSL2/local Ubuntu, you will find the Gazebo Scene Description File (SDF) "**coursework_rf.sdf**" in Moodle. This file sets up the scene in Gazebo containing the FR3 robot and 6 boxes.

- For ConstructSim, we have created a new ROSject for the coursework, called **RF-coursework**. As before, make sure to rename your ROSject with your student ID, as we may use this ROSject to mark the coding aspect and deployment. You will find instructions in the welcome notebook about where you can find the Gazebo Scene Description File (SDF), "**coursework_rf.sdf**". This file sets up the scene in Gazebo containing the FR3 robot and 6 boxes.

What should be submitted?

You should ensure that your ROS node/package addresses the following requirements:

Deployment

- Two ROS packages that group the nodes you used for this coursework and a launch file.
  - Here, you should create a new ROS package for your solution and ensure you can run **colcon built** and use **ros2 run**.

- Your ROS packages can be deployed on the marker's environment, or you can share your ROSject (identified with your student ID).

Functionality

- The ROS nodes allow the FR3 robot to pick and place boxes, and Baxter can stack at least 4 boxes.

- Your ROS node must cover the task specification defined in the Introduction section. Specifically, we will be looking for the following:

- The robot should be able to locate boxes, pick and place boxes, and be able to stack them (1 ROS node)
- Make use of MoveIt! for Motion Planning
- The robot must stack at least 4 boxes – the stacking location should be pre-defined, and you are free to choose the location.
- You have provided a launch file that starts the Gazebo simulation using the SDF file provided.

- **Video demonstration of your solution. For this, use your mobile phone's video recording capabilities to record your computer's screen directly.**
  - One video per group can be shared among members
  - **Your name or the name of your group mates (including student IDs) and the date should be visible in the video recording. You can use Notepad to write this and have it next to the Gazebo. If in doubt, ask us! Edited videos will not be accepted and will result in losing marks.**
  - Failing to submit as specified will result in losing 1 mark for functionality and 1 mark in question 1 in the following section.

Reflective Analysis (aka Lessons Learned)

Individually, you should submit your code and video submission and answer 3 questions that consist of a reflective analysis of your solution. For this, you will have to access the "Assessed Coursework 2025" quiz activity on the course's Moodle page. Below, you can find the questions in the quiz:

1. Write down your groupmates' student IDs and upload your code and video demonstration.
   - Here, you will have a text field where you need to input your groupmates' IDs and, if you are submitting a ZIP file, a drag-and-drop field where you can upload a zip file containing the source code and video demo.
   - For a ConstructSim ROSject, make sure to download your project and submit the ZIP file and also provide the name of your ROSject.

   [2 marks]

2. Explain your train of thought while solving the coursework (2 marks) and the common mistakes that you made (2 marks)

   [max. 200 words, 4 marks]

3. How would you extend your coursework solution? Elaborate on "why you are proposing this solution" (4 marks), and give 2 examples of how your proposed extension will improve the ability of the robot to stack cubes (2 marks per example).

   [max. 400 words, 8 marks]

4. Which behaviour-based control would you choose for this coursework? (1 mark) Why do you choose this behaviour-based control? (2 mark) Which behaviours and how many would you implement? (write a list, 3 marks)

   [max. 300 words, 6 marks]

We recommend using a word processor to complete the above questions and copying/pasting your answers to Moodle. Code should not be included in your answers; this is a reflective analysis rather than a coding exercise. Please follow the University's guidelines about Generative AI:

https://www.gla.ac.uk/myglasgow/sld/ai/students/

From experience, we have found that generative AI-like answers tend to be generic and provide very weak arguments. Hence, you can potentially lose all marks.

Tips:

- You **should** use your video submission to support your statements.
- Note that each question has a maximum word requirement. If your answer is just above this maximum (i.e. 10% more), you will not be penalised as long as you have addressed the question!
- Make sure to acknowledge your group in question 1; otherwise, your submission will be counted as plagiarism and will be reported.
- **All questions should be answered individually and not as a group; otherwise, your submissions will be considered plagiarism and will be reported.**
- The last question follows the format and style of the exam!

How to submit

You should submit via the quiz activity in Moodle:

- A ZIP or 7z file of your ROS package and video demonstration.
    - Note that the video demonstration should be within your ZIP or 7z file.

- **Individual:** The answers to the above questions.

If you have used any external sources, be sure to acknowledge them in your answers. You have unlimited attempts. **We will only grade and give feedback on your last submission.**

Appendix 1: Marking scheme

The following marking scheme is intended to give a broad indication of how marks will be allocated. For details, please refer to this handout's "What should be submitted" section.

Category	Marks
Deployment ROS packages are well structured	0.5
Functionality ROS nodes enable the robot to pick and place boxes, and the robot can stack at least 4 boxes. We will use your video submission as evidence that your solution worked at least once.	1.5
Reflective Analysis Questions Answers should reflect a clear understanding of what the student has developed. See questions 2 to 4 in the “Reflective Analysis” section.	18
Total	20