**Faculty of Technology – Coursework Specification 2020/21**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Module name:** | Intelligent Mobile Robotics | | | | | | |
| **Module code:** | IMAT5233 | | | | | | |
| **Title of the Assignment:** | Creating a 2D Map of an Unknown Environment | | | | | | |
| **This coursework item is:** (delete as appropriate) | | | Summative | | ~~Formative~~ | | |
| **This summative coursework will be marked anonymously** | | | | ~~Yes~~ | | | No |
| **The learning outcomes that are assessed by this coursework are:**   1. A critical awareness of current techniques used for mapping and localisation 2. A comprehensive understanding of stochastic techniques for robot mapping and localisation. 3. Demonstrate originality in the application of knowledge of robot mapping and localisation. | | | | | | | |
| **This coursework is:** (delete as appropriate) | | Individual | | | ~~Group~~ | | |
| **This coursework constitutes** **40 %** **of the overall module mark.** | | | | | | | |
| **Date Set:** | **29th Feb. 2021** | | | | | | |
| **Date & Time Due:** | **Friday 26 March 2021 at noon** | | | | | | |
| **Your marked coursework and feedback will be available to you on:**  If for any reason this is not forthcoming by the due date your module leader will let you know why and when it can be expected. The Head of Studies (headofstudiestec@dmu.ac.uk ) should be informed of any issues relating to the return of marked coursework and feedback.  **NOTE** that you should normally receive feedback on your coursework by **no later than 20 University working days after the formal hand-in date,** provided that you have met the submission deadline. | | | | | | **23rd April 2020** | |
| **When completed you are required to submit your coursework via:**   1. Turnitin for the report 2. Online submission area for zipped folder of code files and video files | | | | | | | |
| **Late submission of coursework** **policy:** Late submissions will be processed in accordance with current University regulations which state:  *“the time period during which a student may submit a piece of work late without authorisation and have the work capped at 40% [50% at PG level] if passed is* ***14 calendar days****. Work submitted unauthorised more than 14 calendar days after the original submission date will receive a mark of 0%. These regulations apply to a student’s first attempt at coursework. Work submitted late without authorisation which constitutes reassessment of a previously failed piece of coursework will always receive a mark of 0%.”* | | | | | | | |
| **Academic Offences and Bad Academic Practices:**  These include plagiarism, cheating, collusion, copying work and reuse of your own work, poor referencing or the passing off of somebody else's ideas as your own. If you are in any doubt about what constitutes an academic offence or bad academic practice you must check with your tutor.  Further information and details of how DSU can support you, if needed, is available at:  [http://www.dmu.ac.uk/dmu-students/the-student-gateway/academic-support-office/academicoffences.aspx](http://www.dmu.ac.uk/dmu-students/the-student-gateway/academic-support-office/academic-offences.aspx) and  [http://www.dmu.ac.uk/dmu-students/the-student-gateway/academic-support-office/bad-academicpractice.aspx](http://www.dmu.ac.uk/dmu-students/the-student-gateway/academic-support-office/bad-academic-practice.aspx) | | | | | | | |

|  |  |
| --- | --- |
| **Tasks to be undertaken:**  **Resources:**  For this assignment you will be using the TurtleBot bot simulator and RoS python libraries or PeopleBot robot (MobileSim simulator) and the Aria C++ libraries. A program implementing a simple wandering behaviour has been created for you – use this as your starting point. You may only use the robot’s odometer and sonar sensors to construct the map. The robot’s laser range finder (SICK) and the Aria Navigation Libraries (SONARNL) should not be used.  **Assignment Specification:**  You must write a program which creates a 2D map of the robot’s environment (a small room or arena). The map should take one of the following forms:   * A set of 2D points (co-ordinates). * A set of line segments. * An occupancy grid.   It must be possible to view the map in one or more of the following ways:   * As a scatter plot (Matlab, excel or openoffice) * As an image (bmp, jpeg, png, ps, pdf) * As a real-time image using an appropriate graphics library (DirectX, OpenGL SFML, FLTK)   Your report should outline how your program constructs a map, giving all your working with diagrams and screenshots. You should also specify any additional external libraries you have used. It should include the following sections:   1. Introduction 2. Map Construction Technique 3. Software Implementation (basic overview – not a code listing) 4. Testing and Results 5. Conclusions 6. Bibliography | |
| **Deliverables to be submitted for assessment:**  A report of no more than 6 pages (not including figures or graphs) must be submitted through the Assignment 1 - Turnitin link under the Assessments tab on Blackboard.  Your software source code should be put in a zip file (please delete \*.sdf and the debug folders) and submitted through the link on BlackBoard. This zip file must be less than 250 Mb.  You may also submit a video or no more than 2 minutes showing your robot’s mapping behaviour but this is not compulsory.  Non-submission of any of the above may lead to a significant reduction of marks and potentially a fail.  **Please note the weighting of these topics as indicated in the marking scheme.** | |
| **How the work will be marked:**  **PLEASE SEE MARKING SCHEME AT THE END** | |
| **Module leader/tutor name:** | **Dr Aboozar Taherkhani** |
| **Contact details:** | aboozar.taherkhani@dmu.ac.uk |

**Marking Scheme**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MARKS** | **Fail < 50%** | **Pass 50% to 59%** | **Merit 60% to 69%** | **Distinction 70% to 100%** |
| **Software**  Worth 80% of the overall mark | Program cannot build a map | Program outputs a set of 2D co-ordinates to the console or a file.  When plotted these co-ordinates give a fair approximation of the robot’s environment. | Program outputs a set of 2D co-ordinates and/or line segments which are drawn on the screen in real-time.  Co-ordinates/line segments give a good approximation of the robot’s environment. | Program constructs an occupancy grid to represent the surrounding environment.  Some method of visualising this occupancy grid is given.  The occupancy grid gives a good approximation of the robot’s environment. |
| **Report**  Worth 20% of the overall mark | Lack of understanding of how maps are built | Brief explanation of the approach taken to build the map is given.  Poor referencing of other work. | Good explanation of the approach taken to build the map is given.  Most working detail is given, perhaps not enough to reproduce the approach. Fair referencing of external sources. | Good explanation of the approach taken to construct the map is given.  Enough working detail is given that the approach could be reproduced by an expert. Good referencing of external sources. |

**Notes:**

* **Failure to prove ownership of any part of the work may lead to a fail.**
* **If you have any questions about this PLEASE ASK**