Untitled-1**SCHOOL OF ENGINEERING AND COMPUTER SCIENCE  
Computer Science**

**AssessmenT DESCRIPTION 2019/2020**

**(Exam tests WORTH ≤15% and Coursework)**

**MODULE DETAILS:**

|  |  |  |  |
| --- | --- | --- | --- |
| Module Number: | 600093 | Trimester: |  |
| Module Title: | Computational Science | | |
| Lecturer: | CK | | |

**COURSEWORK DETAILS:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Assessment Number: |  | of | | |  | | | |
| Title of Assessment: | Simulation and intelligent tracking of a robot | | | | | | | |
| Format: |  |  | | | | |  | |
| Method of Working: |  | | | | | | | |
| Workload Guidance: | Typically, you should expect to spend between | 25 | | and | | 40 | | hours on this assessment |
| Length of Submission: | This assessment should be **no** more than:  *(over length submissions* ***will be*** *penalised as per University policy)* | | **words**  *(excluding diagrams, appendices, references, code)* | | | | | |

**PUBLICATION:**

|  |  |
| --- | --- |
| Date of issue: | 20 Feb 2019 |

**SUBMISSION:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ONE copy of this assessment should be handed in via: |  | | If Other  (state method) |  |
| Time and date for submission: | **Time** | 2pm | **Date** | 1st April 2020 |
| If **multiple hand–ins** please provide details*:* | A series of demonstrations will be scheduled prior to the fnal submission. Please  see the announcement for the times | | | |
| Will submission be scanned via TurnitinUK? |  | If submission is via TurnitinUK, these should be one of the allowed types e.g. Word, RT, PDF, PPT, XLS etc.  Specify any particular requirements in the subumission details  Students MUST NOT submit ZIP or other archive formats. Students are reminded they can **ONLY** submit **ONE** file and must ensure they upload the correct file. | | |
|  | | | | |

The assessment must be submitted **no later** than the time and date shown above, unless an extension has been authorised on a *Request for an Extension for an Assessment* form:

search ‘student forms’ on [https://share.hull.ac.uk](https://share.hull.ac.uk/).

Canvas allows multiple submissions: only the **last** assessment submitted will be marked and if submitted after the coursework deadline late penalties will be applied.

**MARKING:**

|  |  |
| --- | --- |
| Marking will be by: |  |

**ASSESSMENT:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| The assessment is marked out of: | 100 | and is worth |  | % of the module marks |
| **N.B** If multiple hand-ins please indicate the marks and % apportioned to each stage above (i.e. Stage 1 – 50, Stage 2 – 50). It is these marks that will be presented to the exam board. | | | | |

**ASSESSMENT STRATEGY AND LEARNING OUTCOMES:**

The overall assessment strategy is designed to evaluate the student’s achievement of the module learning outcomes, and is subdivided as follows:

|  |  |  |
| --- | --- | --- |
| LO | Learning Outcome | Method of Assessment  *{e.g. report, demo}* |
| ***LO1***  ***LO2***  ***LO3***  ***LO4*** | *Analyse and locate the origins of numerical errors and inexactness in computational approximations*  *Critically analyse algorithms for the implementation of differential equations in simulation applications.*  *Apply an understanding of numerical procedures in the modelling and simulation of real-world applications*  *Explain the concept of modelling and the use of abstraction that allows the use of a machine to solve a problem..* | Report+Demo  Report  Report+Demo  Report+Demo |

|  |  |  |
| --- | --- | --- |
| Assessment Criteria | Contributes to Learning Outcome | Mark |
| Part I  Part II  Part III | LO1, LO2  LO2  LO1, LO3, LO4 | 40  10  50 |

**FEEDBACK**

|  |  |  |  |
| --- | --- | --- | --- |
| Feedback will be given via: |  | Feedback will be given via: |  |
| Exemption  (staff to explain why) |  | | |
| Feedback will be provided no later than 4 ‘teaching weeks’ after the submission date. | | | |

This assessment is set in the context of the learning outcomes for the module and does not by itself constitute a definitive specification of the assessment. If you are in any doubt as to the relationship between what you have been asked to do and the module content you should take this matter up with the member of staff who set the assessment as soon as possible.

You are advised to read the **NOTES** regarding late penalties, over-length assignments, unfair means and quality assurance in your student handbook, which is available on Canvas.

In particular, please be aware that:

* Up to and including 24 hours after the deadline, a penalty of 10%
* More than 24 hours and up to and including 7 days after the deadline; either a penalty of 10% or the mark awarded is reduced to the pass mark, **whichever results in the lower mark**
* More than 7 days after the deadline, a mark of zero is awarded.
* The overlength penalty applies to your written report (which includes bullet points, and lists of text. It does not include contents page, graphs, data tables and appendices). 10-20% over the word count incurs a penalty of 10%. Your mark will be awarded zero if you exceed the word count by more than 20%.

Please be reminded that you are responsible for reading the University Code of Practice on Academic Misconduct through the Assessment section of the Quality Handbook (via the SharePoint site). This govern all forms of illegitimate academic conduct which may be described as cheating, including plagiarism. The term ‘academic misconduct’ is used in the regulations to indicate that a very wide range of behaviour is punishable.

In case of any subsequent dispute, query, or appeal regarding your coursework, you are reminded that it is your responsibility to produce the assignment in question.

**Coursework Description**

This ACW deals with modelling and simulating an intelligent robotic tracking system, and consists of 3 parts.

In part I you will be simulating an ideal robot in one dimension.

The next part, you will make this simulation more realistic, by adding white noise (random numbers with zero mean and a standard deviation, with a normal distribution). This is what is implemented in robot simulation software.

The final part is to train an intelligent agent to track this robot.

There will be a series of compulsory demonstrations prior to the final submission of the report. This demonstration will be feedback on the work, and also enables you to make final corrections based on yoru feedback.

**Part I**

Consider a mobile robot. This robot gets a series of commands to move to a set of coordinates, from its current coordinates in 1D space. (we can assume that this is the floor). A model of this robot is given by the following simplified model

where X is a set of generalised coordinates as a distance from the origin. U changes with time, and represents the distance from the origin the robot has to travel, and is given by

where t is time, and you can take the units to be seconds.

Write a program to simulate this robot. You will need to use different step sizes h for this. Confirm the limiting value of h (see slides from week 2) with the simulation.

Write to file the following, x(k) and U(k), where k is the sample number. You should take your sample interval to be 0.1 seconds. If h=0.01 seconds, then you should write to file the value every 10th integration (sample interval= h\*(number of steps)).

For this part you must

1. Produce the code to simulate the system. Include the logic for changing U over the period of the simulation [15+5marks]
2. The report for this part of the ACW should include results for different values of step size. Look at the upper limit and lower limit for the step. Confirm this using the theoretical discussion done in class (week 2). [20 Marks]

This part has 40 marks,

**Part II**

Assume there is another robot in the vicinity. This second robot has a camera, which uses an intelligent predictive agent to locate and predict the location of the robot from part I. Such systems are often effected by uncertainty and noise. For this part, we can consider all of this together as a random process, and add this to the values you have written to the file in Part I to simulate the noise.

For Part II you will need to simulate noise as a random process with mean μ=0.0 and a standard deviation σ=0.001. You should do this using the Box-Muller method of generating random numbers in a normal distribution. (See slides from week 3).

Add these numbers to the values of x(t) you have in the file from part I.

This part has 20 marks.

These marks are distributed as follows

1. The code for the random number generation, and adding to the data – to simulate the disturbances. (you need to plot the data) [5 Marks]
2. Write up on the random number generation process. You should include a discussion on how and why two normally distributed random numbers are generated starting from one random number in a uniform distribution. This should be approximately 200 words for this (excludes equations etc) [5 Marks]

**Part III**

In this part you will need to develop the intelligent agent which predicts the location of the robot from Part I.

Here the the problem is essentially one of being able to predict the next position of the robot. You will need to use the sensor readings, the values of x(t) after noise has been added for this part. The agent has a single neuron which is capable of learning these movements. You have to write a program to train the neuron.

You would need to do the following

1. Write the code for the intelligent agent (The basic outline of the code was given to you in lectures 2 and 3.)
2. Start with a perceptron and show that the perceptron is not capable of learning this data. You need to explain why.
3. Replace the activation function with a logistic sigmoid. Show that the neuron is now capable of learning. Explain why. (see slides from lectures 2/3)
4. Your write-up must include the program code, the diagrams for modeling the neuron(s), and any other information useful for your conclusions.

You will also need to explain how you would test the agent for unseen data, i.e explain whether it is able to generalize or not.

1. In the data you have generated there are a number of step changes. You should test you training by plotting X(n) and X\_p(n) (to get similar plots as above). You will find that there is a lag – consider why this is the case and report on this in your submission;
2. Discuss possible ways you could predict both position and velocity, either using a perceptron (or a set of perceptrons) or a multilayered network (you need not implement this part (f)).

This part has 40 Marks.

Assessment Marks breakdown (indicative)

|  |  |  |
| --- | --- | --- |
| Parts (a, b, c) | Demo+ report | 20 Marks |
| Parts d, e, | Report | 20 marks |
| Part f | Report | 10 marks |

***Some hints for PART III: (you need not follow these pointers)***

1. Load file into an array – say X(n)
2. If the perceptron has two inputs – then for the first iteration the inputs are 0, and X(1) the output is the prediction X\_p(2) and error = X(2)-X\_p(2)

For the next iteration inputs are X(1), X(2) output is X\_p(3) e= X(3)-X\_p(3)

1. If you have three inputs the same is repeated - and in the first iteration the input vector is 0, 0, X(1)

You can visualize the process in the following table (with 3 inputs)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Iteration | Input(1) | Input(2) | Input(3) | Output\_of\_perceptron | error |
| 1 | 0 | 0 | X(1) | X\_p(2) | X(2)-X\_p(2) |
| 2 | 0 | X(1) | X(2) | X\_p(3) | X(3)-X\_p(3) |
| 3 | X(1) | X(2) | X(3) | X\_p(4) | X(4)-X\_p(4) |
| 4 | X(2) | X(3) | X(4) | X\_p(5) | X(5)-X\_p(5) |
| 5 | X(3) | X(4) | X(5) | X\_p(6) | X(6)-X\_P(6) |