**Module Code:** CS2PJ20

**Assignment report Title:** Robot Simulation + GUI

**Date (when the work completed):** Thursday 14th of December ( extended Deadline)

**Actual hrs spent for the assignment:** 50 Hours more or less

(REPORT TEMPLATE)

Robot GUI Simulation

# Introduction and Showcase

This is a basic simulation of Robots. This simulation is created with javafx, which is helping us to create an interface for the arena of the robots. Added buttons such as start and stop of the simulation and menu bars so the user can save, load or create a new arena with their desire dimensions. Each robot on the arena have their own different characteristics.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **Fig. 1:** Start of the robots simulation |  | **Fig. 2:** Details of th robots on the right |
|  |  |  |
| **Fig. 3:** Menu Bar options |  | **Fig. 4:** About information |

# OOP design diagram

|  |
| --- |
|  |
| **Fig 5.** Class Diagram of Robots |

This is the diagram of the Robot\_GUI\_Simulation, is an application written in java which provides a interface to simulate and interact with robots in the arena. This project works with OOP ( Object-oriented principles) .

* MainRobotSumGUI – Application Folder wher the application launches.
* RobotArena – Manage the arena, robots and interactions.
* Robot (Abstract) – it is the skeleton of all the robots, common properties like drawing, checking collisions and adjustments.
* GameRobot – The robot game, which extends robots
* PacManBot – Robot that consumes Robots within a certain range.
* MyCanvas – Graphical circles, lines, and text
* RobotInterface - User GUI with menus, and buttons to interact with RobotArena.

This are the classes that creates a simulation where the robot move, interact and displays their data on a canvas. GUI allows the user to use buttons such as:

* Load / Save
* Play / Stop
* Create New Arena
* Add Robots

Then the robot that is special is PacManBot which is eating GameRobots when they are in front of it.

# Discussion

A short description of your simulation and potential improvements and other specific extensions you could add.

Robot Simulation GUI is an application where the user can visualize and interacts with robots in the arena. The simulation shows the behavious or the robot such as collision detection, a unique feature like PacManBot eating GameRobots. all the implementatios are very strong but it can be better extensions and mprovements in some ways.

* Tried to add new feature for the robots but the testing were fails ex:
  + Obstacle avoidance.
  + Following a robot.
  + Moving in certain angles.
* Create different arenas, give the user the option to add new shapes for the robots
* User able to customize the robot attributes
* Real-time changes

By appliying this it could be improve the simulation massively, however, it is out of my knowledge yet, in the future I could update this and improve it.

# Reflection

A max 50 words summary reflection on the program you have produced, how you set about developing it, and how you could do such a project better.

# References

(APA/Harvard Style of reference is accepted)

THIS IS PAGE 4

END OF YOUR COURSEWORK PAGE LIMIT, OTHER THAN APPENDIX WITH YOUR EVALUATION

# Appendix – Self Evaluation

Fill in the table below, indicating in column 3 the marks you feel you should get for the different categories, a brief justification in column 4, and the total marks for the section in the Total column.

The marker will read the report, check code and watch the video and determine the actual mark

|  | Student’s own assessment | Mark | Brief justification | Total |
| --- | --- | --- | --- | --- |
| 1. | **Appearance**   * Professional looking 0-2 * Has About and Help 0-2 * Understandable 0-2 * Toolbar buttons 0-2 * Menu 0-2 |  | Comments | 0-10 |
| 2. | **Interaction / Animation.**   * Animation works well 0-1 * Animation controlled by menu 0-1 * Stop/Start by buttons 0-2 * User can add new items 0-2 * User can select item to move 0-2 * User can select item to delete 0-2 |  | Comments | 0-10 |
| 3. | **File Handling etc**   * New arena works 0-2 * Default has populated arena 0-2 * File Load works 0-3 * File Save works 0-3 |  | Comments | 0-10 |
| 4. | **Arena**: does it support different entities   * Robots shown with wheels etc 0-2 * Robots move 0-2 * Has obstacles 0-2 * Has other non moving objects 0-2 * Robot with ‘bump’ type sensor 0-2 * Robot with ‘whisker’ or similar 0-2 * Robot with ‘beam’ sensors 0-2 * Other (specify) 0-2 * Other (specify) 0-2 * Other (specify) 0-2 |  | Number of different types in arena:   |  | | --- | |  |   Comments: | 0-20 |
| 5 | **On Code**   * Appropriate use of names 0-3 * Consistent indentation 0-2 * Has inline comments 0-2 * Has Javadoc comments 0-3 * Correct use of access modifiers 0-3 * Has abstract class for arena items 0-2 * Has good use of inheritance 0-5 |  | Comments | 0-20 |
|  | Section below is assessed by marker of work |  |  |  |
| 6 | Quality of the submitted **report, etc**.  Tests and discussion in report  Video, Javadoc, code, etc., submitted.  Novel features |  | **Evaluation based on submitted work.** | 0-30 |