**Toy Robot Simulator**

|  |  |  |
| --- | --- | --- |
| **Sr No** | **Section** | **Description** |
|  |  |  |
| 1 | Introduction | The project is to code and enable a ‘Robot’ such that it can be positioned on an imaginary table of specific size. It also involves making the Robot move in all directions within the table making sure that the Robot does not fall off the table. |
| 2 | Requirements | -The table should be square shaped and should have specific size  -The Robot should be positioned on the table before executing any other command.  -The robot should be able to move Left and Right i.e change direction and can be positioned in any of the directions e.g. East, West, North & South.  -The position of the robot should be displayed to the user on the screen when demanded  -The system should allow user to exit from the system. |
| 3 | Solution Specification: | The coding is done in C++.  The code will allow a user to specify the table size as per the user requirement. In order to keep it limited a cap of 100 units has been built in the system. In case the user enters junk values while specifying the size, system creates a square table of default size 5 units.  Once the table is created system will prompt the user to initially position the robot on the table asking user to enter the x and y coordinates along with the direction in which the robot should be facing  Once the robot is placed user can move and change direction of the robot on the table using the following commands MOVE, LEFT, RIGHT, REPORT and EXIT.  The solution allows user to use a PLACE command anytime after the initial placing of the robot. All the commands are designed to be case insensitive. |
| 4 | Test | Prepared a test scenarios and test cases. Developed test data and expected result. Recorded the entered data and test results.  Following documents were used for testing:  -Testing document &  -Test data |
|  |  |  |

**Current Restrictions & Possible Enhancements:**

|  |  |
| --- | --- |
| **Restrictions** | **Enhancements** |
|  |  |
| The table is a Square table | The table can be rectangular and/or Square |
| Robot position is displayed to the user as coordinates (x,y). | The position and movement of the robot can be shown graphically to the user. |
| The movement of robot is done by typing commands | The movement of the robot can be done with the help of arrow keys. |
| There is only one robot on the table | Two or more robots can be placed on the table and some sort of game can be devised |
| There are no obstructions on the table | Obstructions can be created at run time and can help in setting up levels of game kind of application. We can crate obstructions on the table and make the robot evade the same. |
| Default values are defined in code | Values can be configured by placing them in a config file which can be read at run-time there by making it a configurable application |
| Manual testing | Automated testing |