**Robocode QuickStart Guide**

**Rules of Battle**

**Your robot**

you have a robot. It can:

* + - Move in any direction
    - Rotate its gun
    - Rotate its radar
    - Fire its gun

**Winning a battle**

You robot must store the highest points over a number of rounds. Points are awarded for:

* + - Surviving - every time someone else dies and you are still alive you get points. You get a bonus for being the last surviving robot.
    - Damage to other robots. You get points for:
      * + Hitting with bullets (1 point for each damage point)
        + Kill by bullets (20% bonus of all damage done by you)
        + Ramming (2 points for each damage point)
        + Kill by ramming (30% bonus of all damage done by you)

You robot has energy. When you run out of energy you are dead.

* + - You lose energy when you hit a wall (the faster you hit the more you lose)
    - You lose energy when you get hit by a bullet (determined by the power of the bullet)
    - You lose energy when you are rammed (determined by the speed of the ram)
    - You lose energy when you fire your gun (the more powerful the bullet the more energy you lose)
    - You gain energy when your bullet hits another robot (3 x the bullet power)
    - If you knock yourself out by dropping your energy to 0 you will be disabled but not dead

**Building a Robot**

* + - You have been given an Eclipse workspace with the beginnings of a robot
    - The Javadoc for the robocode API is available within the robocode installation directory (/Javadoc)
    - Samples robots are also inside the robocode installation directory (/robots/sample)
    - Your robot must extend the Robot class; not the AdvancedRobot class

**Coding Basics**

* + - Your robot is a thread. When the battle starts the run method will be called
    - You should have a while(true) style infinite loop inside the run method to keep the robot active for the duration of the battle
    - If there were no other robots your robot would keep looping forever doing what you tell it to inside the loop until the battle ends
    - To be useful you must respond to events during the battle by implementing the appropriate event handler. Check the Javadoc for a full list.
    - When an event occurs (such as noticing another robot) execution immediately passes to your event handler until the code inside your handler finishes executing.
    - When it finishes executing your robot will go back to its behaviour defined in the infinite loop

**Robot Handling**

* + - Movement is measured in pixels e.g ahead(100) will move 100 pixels forward.
    - Turning is measured in degrees e.g turnRight(90) will turn your robot 90 degrees to the right
    - The faster you go the slower your turn
    - Bullets have a power - the stronger the bullet the slower it goes ( power can be between 0.1 and 3.0)
    - Your gun has a heat - get too hot and it stops working
    - Your gun and radar can turn independently e.g pointing your gun somewhere will not necessarily detect another robot. Your radar must also be pointing that direction. (check the methods called, for example setAdjustGunForRobotTurn

**Responding to Events**

* + - Implement event handlers to respond to events. If you don’t your robot won’t respond to that event.
    - Every event comes with a lot of information passed in under the Event object. Check the API for the information it can give you.
    - Headings and bearings from the event are initially confusing.
      * + Heading is an absolute angle in degrees with 0 being straight up the screen e.g if me.getHeading() returns 90 you are pointing towards the right of the screen.
        + Bearing is a relative angle from something else to your robot’s heading, or put more simply, it is the number of degrees you must turn right in order to face the something else. See diagram below:

**Debugging**

* + - The easiest way to debug is to use lots of System.out.println(“DEBUG”). These will appear in the main battle log which you can access from the UI.
    - It can be helpful to make your robot’s radar arc visible. Options -> Prefs -> View options -> Visible scan arcs.
    - Use the buttons on the UI to step through a battle ‘tick/turn’