

プログラミングⅡ

Report2

提出日：12月2日(月)

所属：工学部情報工学科

学籍番号：135713B

氏名：天願寛之

1.起動時初期位置から、バトルフィールドの中心に移動して、上を向いて止まる。
作成したプログラム

```
package e13_1;

import robocode.*; /*robocode.*をインポート*/
import java.awt.*; /*java.awt.*をインポート*/

/**
 * E135713_2_1 - a robot by 天願寛之 135713B
 */
public class E135713_2_1 extends Robot
{
    public void run() { /*メインメソッド*/
        double px = getX(); /*現在位置の x 座標を取得*/
        double py = getY(); /*現在位置の y 座標を取得*/
        double w = getBattleFieldWidth()/2; /*中心の x 座標を取得*/
        double h = getBattleFieldHeight()/2; /*中心の y 座標を取得*/
        double x = px - w; /*中心との x 座標の差を取得*/
        double y = py - h; /*中心との y 座標の差を取得*/

        if (x<=0 && y<=0){ /*第3象限での出現*/
            turnRight(getAngle(y,x) - getHeading()); /*中心を向く*/
        }else if (x<=0 && y>=0){ /*第2象限での出現*/
            turnRight(getAngle(x,y) + 90 - getHeading()); /*中心を向く*/
        }else if (x>=0 && y>=0){ /*第1象限での出現*/
            turnRight(getAngle(y,x) + 180 - getHeading()); /*中心を向く*/
        }else if (x>=0 && y<=0){ /*第4象限での出現*/
            turnRight(getAngle(x,y) + 270 - getHeading()); /*中心を向く*/
        }else turnRight(getHeading()); /*中心での出現*/ /*上を向く*/
        ahead(Math.sqrt(x*x+y*y)); /*中心に移動*/
        turnRight(getHeading()); /*上を向く*/
    }

    public double getAngle(double x, double y){ /*getAngle メソッド 引数 x,y*/
        x = Math.abs(x); /*x を絶対値化*/
        y = Math.abs(y); /*y を絶対値化*/
        double angle = Math.atan(y/x); /*中心に対してのラジアンを取得*/
        return (angle*180/Math.PI); /*中心に対しての角度を返す*/
    }
}
```

2.初期位置から、バトルフィールド中心を半径として、バトルフィールド中心を中心として円運動するロボット。このとき、半径がバトルフィールドの短辺に収まらない場合は、中心へ向けて、移動後、円運動すること。

作成したプログラム

```
package e13_1;

import robocode.*; /*robocode.*をインポート*/
import java.awt.Color; /*java.awt.Colorをインポート*/
import robocode.AdvancedRobot; /*robocode.AdvancedRobotをインポート*/

/**
 * MyFirstRobot2 - a robot by 天願寛之 135713B
 */
public class E135713_2_2 extends AdvancedRobot{

    /*メインメソッド*/
    public void run() {

        double px = getX(); /*現在のロボットのx座標を取得*/
        double py = getY(); /*現在のロボットのy座標を取得*/
        double w = getBattleFieldWidth()/2; /*中心のx座標を取得*/
        double h = getBattleFieldHeight()/2; /*中心のy座標を取得*/
        double x = px - w; /*中心とのx座標の差を取得*/
        double y = py - h; /*中心とのy座標の差を取得*/

        if(Math.sqrt(x*x+y*y)<250){ /*出現位置の中心からの距離が250未満の場合*/
            if (x<=0 && y<=0){ /*第3象限での出現*/
                turnRight(getAngle(y,x) + 90 - getHeading()); /*出現位置で中心に対して垂直に向く*/
            }else if (x<=0 && y>=0){ /*第2象限での出現*/
                turnRight(getAngle(x,y) + 180 - getHeading()); /*出現位置で中心に対して垂直に向く*/
            }else if (x>=0 && y>=0){ /*第1象限での出現*/
                turnRight(getAngle(y,x) + 270 - getHeading()); /*出現位置で中心に対して垂直に向く*/
            }else if (x>=0 && y<=0){ /*第4象限での出現*/
                turnRight(getAngle(x,y) + 360 - getHeading()); /*出現位置で中心に対して垂直に向く*/
            }
            Circle(); /*Circleメソッドを参照*/
        }

        if(Math.sqrt(x*x+y*y)>=250){ /*出現位置の中心からの距離が250以上の場合*/
            if (x<=0 && y<=0){ /*第3象限での出現*/
                turnRight(getAngle(y,x) - getHeading()); /*中心を向く*/
            }else if (x<=0 && y>=0){ /*第2象限での出現*/
                turnRight(getAngle(x,y) + 90 - getHeading()); /*中心を向く*/
            }else if (x>=0 && y>=0){ /*第1象限での出現*/
                turnRight(getAngle(y,x) + 180 - getHeading()); /*中心を向く*/
            }else if (x>=0 && y<=0){ /*第4象限での出現*/
                turnRight(getAngle(x,y) + 270 - getHeading()); /*中心を向く*/
            }
            ahead(Math.sqrt(x*x+y*y) - 250); /*中心に向かって(半径-250)移動*/
        }

        if (x<=0 && y<=0){ /*第3象限での出現*/
            turnRight(getAngle(y,x) + 90 - getHeading()); /*移動場所から中心に対して垂直に向く*/
        }else if (x<=0 && y>=0){ /*第2象限での出現*/
            turnRight(getAngle(x,y) + 180 - getHeading()); /*移動場所から中心に対して垂直に向く*/
        }else if (x>=0 && y>=0){ /*第1象限での出現*/
            turnRight(getAngle(y,x) + 270 - getHeading()); /*移動場所から中心に対して垂直に向く*/
        }else if (x>=0 && y<=0){ /*第4象限での出現*/
            turnRight(getAngle(x,y) + 360 - getHeading()); /*移動場所から中心に対して垂直に向く*/
        }
        Circle(); /*Circleメソッドを参照*/
    }
}
```

```

    }

    public double getAngle(double x, double y){ /*getAngle メソッド 引数 x,y*/
        x = Math.abs(x); /*x を絶対値化*/
        y = Math.abs(y); /*y を絶対値化*/
        double angle = Math.atan(y/x); /*中心に対してのラジアンを取得*/
        return (angle*180/Math.PI); /*中心に対しての角度を返す*/
    }

    public void Circle(){ /*Circle メソッドを取得*/
        double px = getX(); /*現在のロボットの x 座標を取得
*/
        double py = getY(); /*現在のロボットの y 座標を取得*/
        double w = getBattleFieldWidth()/2; /*中心の x 座標を取得*/
        double h = getBattleFieldHeight()/2; /*中心の y 座標を取得*/
        double x = px - w; /*中心との x 座標の差を取得*/
        double y = py - h; /*中心との y 座標の差を取得*/
        while(true){ /*advancedrobot を用いて円周の長さを進む間に 360 度回転させる*/
            setTurnLeft(1);
            ahead(2*Math.PI*Math.sqrt(x*x+y*y)/360);
        }
    }
}

```