Assignment Robotics:

Q1) Create a robot to perform square and circular motion without using gear.

a) **Sqaure:**

Code:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package robotics;

//Import packages
import ch.aplu.robotsim.*;

/**
 *
 * @author admin
 */
public class RobotSquare {

   public RobotSquare() {
      TurtleRobot turtleRobot = new TurtleRobot();
      while(true)
      {
         turtleRobot.forward(100);
         turtleRobot.right(90);

      }

   }

   public static void main(String[] args){
      new RobotSquare();
   }

}
```

b) **Circle:**

Code:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package robotics;
import ch.aplu.robotsim.TurtleRobot;

/**
 *
 * @author admin
 */
public class RobotCircle {

   public RobotCircle() {
      TurtleRobot turtleRobot = new TurtleRobot();
      while(true)
      {
         turtleRobot.forward(2);
         turtleRobot.right(2);

      }

   }

   public static void main(String[] args){
      new RobotCircle();

   }

}
```

Q2) **Create a robot to perform rectangle and circular motion without using gear.**

a) **Rectangle:**

Code:

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package robotics;

import ch.aplu.robotsim.Gear;
import ch.aplu.robotsim.NxtRobot;

/**
```

```
 *
 * @author admin
 */
public class RobotGearRectangle {

    public RobotGearRectangle() {
        NxtRobot nxtRobot = new NxtRobot();
        Gear gearBox = new Gear();
        nxtRobot.addPart(gearBox);

        gearBox.setSpeed(100);

        while(true){
            gearBox.forward(2000);
            gearBox.left(280);
        }

    }

    public static void main(String[] args){
        new RobotGearRectangle();
    }
}
```

b) Circle:

Code:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */


package robotics;
import ch.aplu.robotsim.NxtRobot;
import ch.aplu.robotsim.Gear;


/**
 *
 * @author admin
 */
public class RobotGearCircle {

   public RobotGearCircle() {
      NxtRobot nxtRobot = new NxtRobot();
      Gear gearBox = new Gear();
      nxtRobot.addPart(gearBox);

      gearBox.setSpeed(10000);
      while(true){
         gearBox.rightArc(0.5);
      }
   }

   public static void main(String[] args){
      new RobotGearCircle();
   }

}
```

Q3) Program to do square using while or for loop. Change direction based on condition and continue motor movement using switch case.

Code:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package robotics;
```

```
//Import packages
import ch.aplu.robotsim.*;
import java.util.Scanner;
/**
 *
 * @author admin
 */

public class RobotMotorSquare {
    NxtRobot nxtRobot;
    Motor motorLeft, motorRight;

    public RobotMotorSquare() {
        nxtRobot = new NxtRobot();
        motorLeft = new Motor(MotorPort.A);
        motorRight = new Motor(MotorPort.B);

        nxtRobot.addPart(motorLeft);
        nxtRobot.addPart(motorRight);

        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter direction:");
        String direction =  scanner.nextLine();

        switch(direction){
            case "left":
                moveRobot("left");
            break;

            case "right":
                moveRobot("right");
            break;


            default:
                System.out.println("Please specify the correct direction");
                System.exit(1);
            break;
        }
    }


    public void moveRobot(String direction) {
        for(int i = 0; i < 4; i++) {
            if(direction == "left") {
                motorLeft.forward();
                Tools.delay(1090);
                motorRight.forward();
            }
            else {
                motorRight.forward();
                Tools.delay(1090);
                motorLeft.forward();
            }


            Tools.delay(1090);
            motorLeft.stop();
            motorRight.stop();
        }
    }


    public static void main(String[] args) {
        new RobotMotorSquare();
    }
```

```
}
```