**import** java.util.HashSet;

**import** java.util.Set;

**enum** Direction {

    N, E, S, W;

**public** Direction turnLeft() {

**return** values()[(ordinal() + 3) % 4];

    }

**public** Direction turnRight() {

**return** values()[(ordinal() + 1) % 4];

    }

}

**public** **class** MarsRover {

**private** **int** x;

**private** **int** y;

**private** Direction direction;

**private** Set<String> obstacles;

**public** MarsRover(**int** x, **int** y, Direction direction) {

**this**.x = x;

**this**.y = y;

**this**.direction = direction;

**this**.obstacles = **new** HashSet<>();

    }

**public** **void** rotate(**char** rotation) {

**if** (rotation == 'L') {

            direction = direction.turnLeft();

        } **else** **if** (rotation == 'R') {

            direction = direction.turnRight();

        }

    }

**public** **void** move() {

**int** nextX = x;

**int** nextY = y;

**switch** (direction) {

**case** N:

                nextY++;

**break**;

**case** E:

                nextX++;

**break**;

**case** S:

                nextY--;

**break**;

**case** W:

                nextX--;

**break**;

        }

        String nextPosition = nextX + ":" + nextY;

**if** (!obstacles.contains(nextPosition) && isValidPosition(nextX, nextY)) {

            x = nextX;

            y = nextY;

        }

    }

**private** **boolean** isValidPosition(**int** x, **int** y) {

**int** maxX = 10;

**int** maxY = 10;

**return** x >= 0 && x <= maxX && y >= 0 && y <= maxY;

    }

**public** **void** addObstacle(**int** x, **int** y) {

        obstacles.add(x + ":" + y);

    }

**public** String getPosition() {

**return** x + ":" + y + " (" + direction + ")";

    }

**public** **static** **void** main(String[] args) {

        MarsRover rover = **new** MarsRover(0, 0, Direction.N);

        rover.addObstacle(2, 2);

        rover.addObstacle(4, 6);

        String commands = "LMLMLMLMM";

**for** (**char** command : commands.toCharArray()) {

**if** (command == 'L' || command == 'R') {

                rover.rotate(command);

            } **else** **if** (command == 'M') {

                rover.move();

            }

        }

        System.out.println("Final position: " + rover.getPosition());

    }

}