Doris Chen

1. Modify your Robot.java file to simplify the handling of capitalization. Consider converting all input to lower case and only process lower-case text.

```
> aUTo CleaN
Automated Cleaning started
```

2. Process common spelling errors by maintain a list of them and checking key input words against them. Submit the string with the corrected words to the parser.

```
private String[] spellingerrors = { "rite", "wright", "write", "mov", "muv", "op", "doen", "dawn", "cleen", "ledr", "lft" };
```

3. Modify the processing of the semantic graph object to address common grammar errors, such as reordering of words. Cover at least five grammar errors.
    1) Right/left/up/down move please/ right go please
    2) Go you/your to left/right/up/down
    3) 1 grid for right/ grid 1 right go
    4) Robot go go right/left/up/down
    5) I let you to go right
4. When the user issues a command such as “begin record”, the robot executes and records actions until the user issues a command such as “end record.”
5. Following the "end record" command, the system asks the user for a name of the recorded plan. The system will store the plan under that name.
6. When the user issues a command like: “execute plan Bob” the system will execute the current plan from the current position of the robot. Please notice that most likely you have to temporarily deactivate the read-evaluate-return loop, until the plan is executed to its completion. Please notice that you do not have to worry about error recovery, as the system itself will not let you walk into walls or off the side of the grid.

```
> begin record
Recording started
> please go move to the right
Recording path
Got it.
doris is going to go right
> right
Recording path
Processing Single Word
10-4.
Got it! doris will immediately move right
> End Recording
Recording path
Recording finished
Please Enter Your Plan Name:
> A
Named the plan a
> Execute Plan A
Executing plan a
```

This is step 4,5,6 demo

7. Add the ability to recall a plan in which all instructions are symmetric. For example, if the user states: “execute symmetric plan Bob” then the system will execute the stored plan, except that all occurrences of **up** are replaced with **down** and vice versa and all occurrences with **left** are replaced with **right** and vice versa.

```
> Execute symmetric plan A
Executing symmetric plan a
> Execute plan B
Executing plan b
Plan is not recorded
Sorry, could you be more specific?
`
```

Plan B is not in the recording path, and the system will give a notification

8. Instruct the robot to clean up all or a portion of the dirty tiles. Please use your bfsM or astarM procedures. You should be able to specify the following options.
    1. Clean all remaining dirty tiles.

```
> auto clean
Automated Cleaning started
```

    2. Specify a list of tiles by coordinates.

```
> Clean coordinates
Please enter coordinates in the form: (a,b),(c,d)
> (2,3),(3,4),(4,5)
Automated Clean Coordinates if dirty
```

    3. Specify a rectangle of tiles. If the rectangle includes walls, that will be fine, however, your robot should not bang it's head against walls.

```
> clean Rectangle
Please enter coordinates in the form: (upperleft x,upperleft y),(lowerright x,lowerright y)
> (0,0),(6,6)
Automated Clean dirty tile in the rectangle
```

```
> clean rectangle
Please enter coordinates in the form: (upperleft x,upperleft y),(lowerright x,lowerright y)
> (0,0),(1,1)
Automated Clean dirty tile in the rectangle
dsb
> clean rectangle
Please enter coordinates in the form: (upperleft x,upperleft y),(lowerright x,lowerright y)
> (1,1),(2,2)
Automated Clean dirty tile in the rectangle
dsb
> clean rectangle
Please enter coordinates in the form: (upperleft x,upperleft y),(lowerright x,lowerright y)
> (0,0),(6,6)
Automated Clean dirty tile in the rectangle
```