TOY ROBOT LIBRARY & SIMULATION

Shaun Chua
https://github.com/GitHubShaun/ToyRobot

GOAL:

Create a library (java) that can be imported into a toy robot simulation by the user. Library is able to read in PLACE X,Y,DIRECTION, MOVE, LEFT, RIGHT, REPORT commands received by the user's program.
Example usage:

```
...
    public static void main(String[] args){
        ToyRobot robot = new ToyRobot(0,0,NORTH);
        robot.move();
        robot.report();
    }
...
```

My emphasis was to provide an easy implementation for the user so there are additional constructors that the user can choose and pick from for their design.

DESIGNED AND DEVELOPED ON:

macOS: version 10.14.5 (18F132)
IntelliJ IDEA: 2019.1.3, build IC-191.7479.19. Copyright JetBrains s.r.o., (c) 2000-2019
Java : java version "1.8.0_144"
Junit 4: junit-4.13-beta-3

ASSUMPTIONS:

- User has similar system environment,
    - JDK is installed
    - MacOS
    - file hierarchy is unchanged
- User already knows how to use java libraries,
    - adding .jar file as a dependency
    - how to use libraries in their own code
- Test files are in the testfiles directory in the project folder
    - .java simulation file assumes .txt test files ONLY consists of valid commands

COPY OF THE SCRIPT.COMMAND FILE:

This was created to simplify the compile/run stage.

```
echo "ToyRobot Library Simulation Compiling, by Shaun Chua\n..."
javac -d . -cp ".:lib/ToyRobot.jar" src/toyrobot/ToyRobotSim.java
if [ $# -eq 0 ]
then
    echo "No arguments provided.\nUsage: sh testscript.sh [filename.txt]
[filename.txt]"
else
    for i in "$@"
    do
        java -cp "lib/ToyRobot.jar" toyrobot.ToyRobotSim testfiles/$i
    done
fi
echo "...\nCompleted."
```

javac -d . -cp ".:lib/ToyRobot.jar" src/toyrobot/ToyRobotSim.java

Compiles the toy robot simulation using my library design from the lib directory, into the current directory.
if [$# -eq 0]
Shell script is expecting test cases, input as textfiles (see below), as command line arguments.
for i in "$@"
For loop iterating through the input arguments.
java -cp "lib/ToyRobot.jar" toyrobot.ToyRobotSim testfiles/$i
Java virtual machine running the simulation using the test files provided. ToyRobotSim.java will read these files and perform tasks depending on the commands given to the robot.

File Tree:

```
ToyRobot-master/
├── JavaDoc
│   ├── allclasses-frame.html
│   ├── allclasses-noframe.html
│   ├── constant-values.html
│   ├── deprecated-list.html
│   ├── help-doc.html
│   ├── index-files
│   │   ├── index-1.html
│   │   ├── index-2.html
│   │   ├── index-3.html
│   │   ├── index-4.html
│   │   ├── index-5.html
│   │   ├── index-6.html
│   │   └── index-7.html
│   ├── index.html
│   ├── overview-tree.html
│   ├── package-list
│   ├── script.js
│   ├── stylesheet.css
│   └── toyrobot
│       ├── ToyRobot.html
│       ├── package-frame.html
│       ├── package-summary.html
│       └── package-tree.html
├── README.txt
├── dist
│   └── ToyRobot.jar
├── lib
│   └── ToyRobot.jar
```

```
├── script.command
├── src
│   └── toyrobot
│       ├── ToyRobot.java
│       └── ToyRobotSim.java
├── testfiles
│   ├── testcase1.txt
│   ├── testcase2.txt
│   ├── testcase3.txt
│   └── testcase4.txt
└── tests
    └── toyrobot
        ├── ToyRobotCommandsTest.java
        ├── ToyRobotConstructorTests.java
        ├── ToyRobotRESTTest.java
        ├── ToyRobotSimTest.java
        ├── ToyRobotSimTestCase1.java
        ├── ToyRobotSimTestCase2.java
        ├── ToyRobotSimTestCase3.java
        ├── ToyRobotSimTestCase4.java
        └── ToyRobotTest.java
```

ToyRobotSim.java – toy robot simulation program using the library
ToyRobot.java – java library that reads commands and performs actions.
ToyRobot.html – documentation of my implemented library.
ToyRobot.jar - .jar library file that can be used for distribution/importing into programs.

TEST CASES IN THE TESTFILES FOLDER:

testcase1.txt: Test basic functions.
PLACE 0,0,NORTH
MOVE
REPORT

testcase2.txt: Test basic functions.
PLACE 0,0,NORTH
LEFT
REPORT

testcase3.txt: Test basic functions.
PLACE 1,2,EAST
MOVE
MOVE
LEFT
MOVE

REPORT

testcase4.txt: Test commands are not registers until a valid PLACE command is recognized.
MOVE
REPORT
PLACE 0,0,SOUTH
LEFT
REPORT