

Mars Explorers Documentation

Lawrence Righi Boechat

Project Design

The project solves the second problem described in the received instructions for the Amdocs selection process. For this purpose, four java classes were created and developed utilizing Eclipse IDE Version: 2018-12 (4.10.0) Build ID: 20181214-0600 in a machine running Windows 10 Home 64 bits.

The Command class in the Command.java file contains the main method and acts as the interface in which the user sends instructions and receives the outputs required to control the rovers. It was thought as being NASA command center to send and receive information to and from the rovers. It deals with data received in the form of strings from a file or a manual keyboard input. Also, Command has access to the rover current position, making it possible to broadcast this information to the other rover in order to avoid collision. Additionally, Command receives the user desired plateau size, to which it creates a map and sends it to the rovers for border recognition.

The Rovers class is responsible for the rover operation. It receives data from the Command class containing: it's initial position and direction, it's set of instructions and also the other rover current location. Each rover object has its own position module, responsible to track and set the precise rover location in the Martian plateau. The Rover will then attempt to execute it's set of instructions, avoiding and skipping any that could potentially cause problems such as border trespassing and rover collision. The class also contains an output method of its current location and a method to broadcast it's position directly to the operator terminal.

Position.java contains the class that could be described as the rover tracker. It possess the methods which records and updates the rover position to itself, acting as an independent camera interpreter, as it would be guided and updated through the rover movement.

Lastly, the Plateau class is an interpretation of the plateau as viewed by the rover, maintaining the properties of the locale described by the user.

How to run

The author assumes possession of a Java interpreter, as it was forbidden to send an executable file. The .zip carries all files necessary to run a code from a terminal, as they are already compiled. Extract the folder directly to a terminal accessible location and run java in “MarsRovers.Command”.

Here is an example tested by the author’s windows machine:

- Extract files to C:\

In a cmd terminal window:

- cd C:\Mars Explorers
- java MarsRovers.Command

The program should start by asking if you wish to input data via file or manually.

Inputting via file:

Type 1 and Enter as requested by the program.

Type in the file name along with its extension name. There is an additional Test_Input.txt in the folder containing the given test input from the problem that was used and confirmed in testings. Should the user want to input another file with different set of instructions, insert it on the same folder.

Inputting manually:

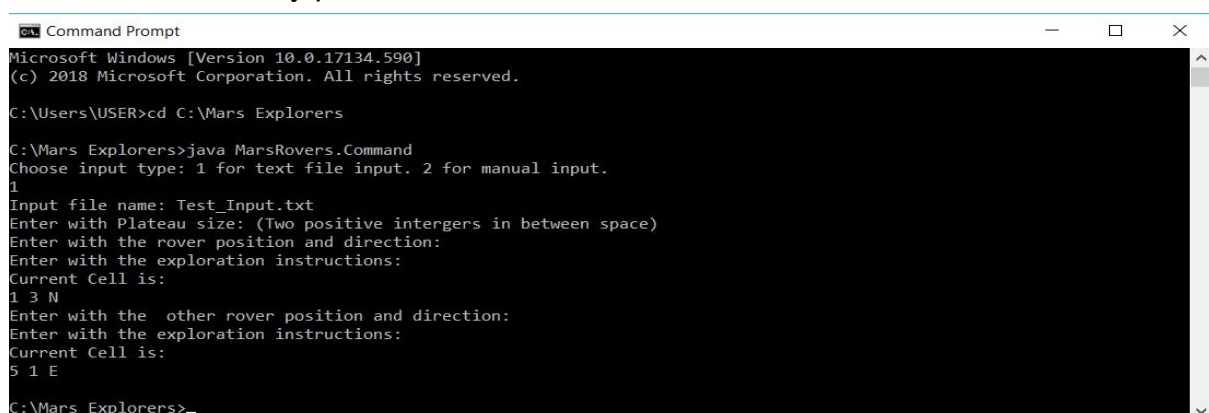
Type 2 and Enter as requested by the program.

The program will guide the user step by step to input the data, as well as printing outcomes and possible issues found on the run time.

Inside the folder there are also scripted Java codes of the package, permitting inspection and evaluation of what was build.

Appendix:

Proof of execution by picture:



```
Microsoft Windows [Version 10.0.17134.590]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\USER>cd C:\Mars Explorers

C:\Mars Explorers>java MarsRovers.Command
Choose input type: 1 for text file input. 2 for manual input.
1
Input file name: Test_Input.txt
Enter with Plateau size: (Two positive intergers in between space)
Enter with the rover position and direction:
Enter with the exploration instructions:
Current Cell is:
1 3 N
Enter with the other rover position and direction:
Enter with the exploration instructions:
Current Cell is:
5 1 E

C:\Mars Explorers>
```