A-Star Search Implementation in JAVA

**Project Description:**

The **node-search-api** project is a simple A-star search algorithm in JAVA, where the program calculates the path between two nodes in a given map. The map here is supplied through a large\_map.txt or small\_map.txt file and it is expected that the program must generate e result\_map.txt file with the highlighted path using #.

**Tools/Technologies Used:**

IDE: Eclipse Kepler

JDK Version: JAVA 1.7

Maven: 3.0.4

Junit: 4.9

Repo: GitHub

**Technical Description:**

Source Code Summary:

The api is built by keeping in mind that this can extended further to add new search algorithms without affecting the existing search logic.

This is a single interface API(NodeSearchInterface) just have single method **findPath** that finds the path in the given map and outputs the result to a file called **result\_map.txt.** The actual implementation is abstracted behind the interface and just does the intended job without exposing any complexities to the end user.

There is a factory class called **NodeSearchFactory** is the entry point to this API, It returns an implementation according to the search type provided by the user.

We tried to use both Single ton pattern and Factory pattern here to organize and optimize the code. The factory pattern here helps in plugin new implementations of search algorithms without affecting the existing search logic. It also enables to plug in implementations of distance calculation formula between nodes other than manhattan formula using **NodeFactory** class.

The end result is being output to a file called **result\_map.txt** defined in the **NodeConstants** class

Unit Test Coverage:

**NodeSearchTest** provides unit test cases to test the available operations of NodeSearchInterface . All the required resources file are provided under **src/test/resources** folder of the application.

Please refer to the instructions in README.txt file to run and test the application.

GitHub Location : https://github.com/antaryami/search-algorithms.git

NOTE: A sample result\_map.txt file has been provided under document section, it contains the map highlighted with the symbols # and it is also numbered with the sequence for better understandability.