The Project consist of:

- Code Implementation of search algorithms
- Jar file for execution
- Sample Input files
- ReadME.
- GUI code

Software's Needed:

- JDK16.
- XAMPP

Instructions to setup XAMPP Server:

To execute GUI code, we need XAMPP server. We need to install the XAMPP (Links to an external site.) web server, which includes the Apache http web server, PHP, MySQL (MariaDB), and PHPMyAdmin (these are the only components we need). It's about 125MBs (775MBs after installation) and can be installed on Windows, Linux, and OS X. The installation directory is \xampp for Windows, /opt/lampp for Linux, and /Applications/XAMPP for OS X. To start the server on Windows, run \xampp\xampp-control.exe and start the Apache web server. You may have to change the Security properties of this executable to Full Control for Users. Can test the project on your PC/laptop using the Chrome/Mozilla Firefox web browser.

Instructions to execute the code:

Execute through GUI:

Download the zip file. Unzip it inside your web server document root directory (ie, inside the htdocs sub-directory in the XAMPP instalation directory). On Linux, you may have to do this as the root user.

URL to test in Chrome/Firefox:
http://localhost/DAASearchProject/SearcAlgo.php

Search button will return size of the input, type of algorithm, search element and their index and run time of the algorithm. Compare button will return the same along with that it will generate the graph with input size vs run time.

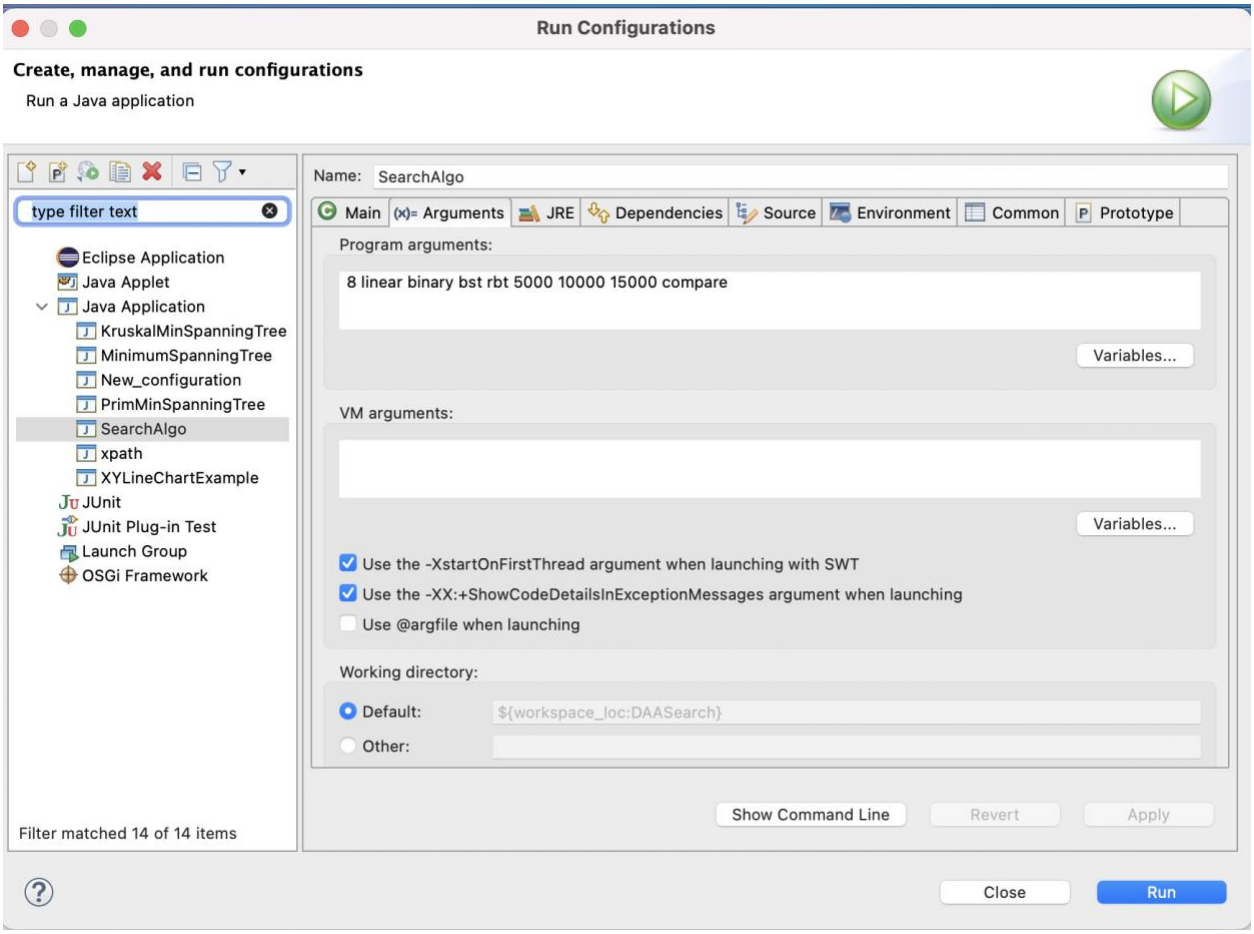
Execute through Terminal:

Command Line argument:

Searching: java -jar project2Daa.jar <Input Search Element> <size1> <size2> <size3> <size4> <…> <sizeN>
Example:
java -jar project2Daa.jar 8 1000 2000 3000 4000
Comparing all algorithms: java -jar project2Daa.jar <Input Search Element> <size1> <size2> <size3> <size4> <…> <sizeN> compare
Example:
java -jar project2Daa.jar 8 1000 2000 3000 4000 compare

Execute through Eclipse IDE:

You can import the code directly into Eclipse IDE and run the main file SearchAlgo.java by passing the arguments in configuration like “8 linear binary bst rbt 5000 10000 15000” or “8 linear binary bst rbt 5000 10000 15000 compare”.



Note: To generate graph through Eclipse IDE, we need JFree chart library in the classpath which I added in the classpath as well as attached in the files for reference. No need to add JFree library while executing through Jar file.