

## How to run code:

Code should run correctly by running the command:

Mvn clean package

To test:

mvn tests

## Refactor 1:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/9de32e9e05a70c92818650120db2ed73279d9089>

## Refactor 2:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/f2612bc71fc84c11eeb5b94529627d7d83af5d7a>

## Refactor 3:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/f0a9b11ec95cd79f2b220ac21e4deeb254451ae5>

## Refactor 4:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/cd734eaa07b9bd90b22701f6acd9f2ee9676c35>

## Refactor 5:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/45feeada9fa6e3b79dcebd957428c2939388fdc7>

## Abstract Class Implemented:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/844daa85a9ab8cdccca0c987f478fa51371edf13>

I applied the choice for either BFS or DFS by creating an enum, and then adding a parameter in GraphSearch() method which allows me to choose either BFS or DFS.

## Bfs and DFS Implemented

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/6b3522c4e39359eb531d2ccd1efceda19b5069d7>

I implemented the actual Bfs and Dfs logic into their respective classes and verified they still work as intended

## Example output following BFS and DFS Implementation:

```
[INFO] Running DotGraphParserTest
Current nodes in graph: [A, B, C]
Node A removed successfully.
Node C removed successfully.
Skipping removal for non-existing node: E
Nodes A, C, E removed successfully.
Current nodes in graph: [B, D]
Current nodes in graph: [A, B, C, X, Y]
Current nodes in graph: [A, B, C, X]
Node X removed successfully.
Current nodes in graph: [A, B, C]
[main] INFO guru.nidi.graphviz.engine.GraphvizCmdLineEngine - input
file:///var/folders/62/tc_f876x4q3bcr1v_0gmtqlh0000gn/T/GraphvizJava/DotEngine17832235304794628109/dotfile.dot
[main] INFO guru.nidi.graphviz.service.CommandLineExecutor - executing command [/bin/sh, -c, dot -Kdot -Tsvg
/var/folders/62/tc_f876x4q3bcr1v_0gmtqlh0000gn/T/GraphvizJava/DotEngine17832235304794628109/dotfile.dot -ooutfile.svg]
[main] INFO guru.nidi.graphviz.engine.GraphvizCmdLineEngine - output
file:///var/folders/62/tc_f876x4q3bcr1v_0gmtqlh0000gn/T/GraphvizJava/DotEngine17832235304794628109/outfile.svg
Current nodes in graph: [A, B, C]
[INFO] Tests run: 14, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.802 s -- in DotGraphParserTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 14, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- jar:3.4.1:jar (default-jar) @ CSE464_Project ---
[INFO] Building jar: /Users/aryamanmarwaha/IdeaProjects/CSE-464-2024-amarwah3/target/CSE464_Project-1.0-SNAPSHOT.jar
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 2.454 s
[INFO] Finished at: 2024-12-03T23:09:55-07:00
[INFO]
```

## Implemented RandomWalkGraphSearch as well as updated test cases for input.dot

<https://github.com/hibobshin/CSE-464-2024-amarwah3/commit/5c2ee0d51742ef472142359690e26cb8773c0abe>

Here I implemented RandomWalkGraphSearch, including making a new file. I then updated DotGraphSearch to be able to handle RandomWalk as an enum, and allowed the parameter to be accepted in an input. I then updated all the test cases to incorporate input.dot from canvas.

## Example output for RandomWalkGraphSearch:

```

[INFO] Running DotGraphParserTest
Current nodes in graph: [a, b, c, d, e, f, g, h]
Node a removed successfully.
Node c removed successfully.
Node e removed successfully.
Nodes a, c, e removed successfully.
Current nodes in graph: [b, d, f, g, h]
Current nodes in graph: [a, b, c, d, e, f, g, h, x, y]
Random Walk Testing
Visiting a -> b
Visiting a -> b -> c
Random Walk Path [1]: a -> b -> c
Visiting a -> e
Visiting a -> e -> g
Visiting a -> e -> g -> h
No unvisited neighbors available. Random walk terminates.
Random Walk Path [2]: a -> e -> g -> h
Random walk did not reach the destination node 'c' in this run.
Visiting a -> b
Visiting a -> b -> c
Random Walk Path [3]: a -> b -> c
Visiting a -> e
Visiting a -> e -> f
Visiting a -> e -> f -> h
No unvisited neighbors available. Random walk terminates.
Random Walk Path [4]: a -> e -> f -> h
Random walk did not reach the destination node 'c' in this run.
Visiting a -> b
Visiting a -> b -> c
Random Walk Path [5]: a -> b -> c
Current nodes in graph: [a, b, c, d, e, f, g, h, x]
Node x removed successfully.
Current nodes in graph: [a, b, c, d, e, f, g, h]
[main] INFO guru.nidi.graphviz.engine.GraphvizCmdLineEngine - input
Current nodes in graph: [a, b, c, d, e, f, g, h]
[main] INFO guru.nidi.graphviz.engine.GraphvizCmdLineEngine - input
file:///var/folders/42/tc-f876x4q3bcr1v_0gmtqlh0808gn/T/GraphvizJava/DotEngine5675515409948720623/dotfile.dot
[main] INFO guru.nidi.graphviz.service.CommandLineExecutor - executing command [/bin/sh, -c, dot -Kdot -Tsvg
/var/folders/62/tc-f876x4q3bcr1v_0gmtqlh0808gn/T/GraphvizJava/DotEngine5675515409948720623/dotfile.dot -ooutfile.svg]
[main] INFO guru.nidi.graphviz.engine.GraphvizCmdLineEngine - output
file:///var/folders/62/tc-f876x4q3bcr1v_0gmtqlh0808gn/T/GraphvizJava/DotEngine5675515409948720623/outfile.svg
Current nodes in graph: [a, b, c, d, e, f, g, h]
[INFO] Tests run: 11, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 1.236 s -- in DotGraphParserTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 11, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- jar:3.4.1:jar (default-jar) @ CSE464_Project ---
[INFO] Building jar: /Users/aryamannmarwaha/IdeaProjects/CSE-464-2024-amarwah3/target/CSE464_Project-1.0-SNAPSHOT.jar
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 3.625 s
[INFO] Finished at: 2024-12-04T12:56:21-07:00
[INFO]

```

## Pull Request Link:

<https://github.com/hibobshin/CSE-464-2024-amarwah3/pull/8>