

Yue Fang ASUID: 1215500923

Java -version : java11

JgraphT -version: 1.5.0

The input file is src/testFiles/dotGraph.dot

You can run the TestCode using the “run” option in IntelliJ IDEA. When using ‘run’, there are four options for four feature-tests.

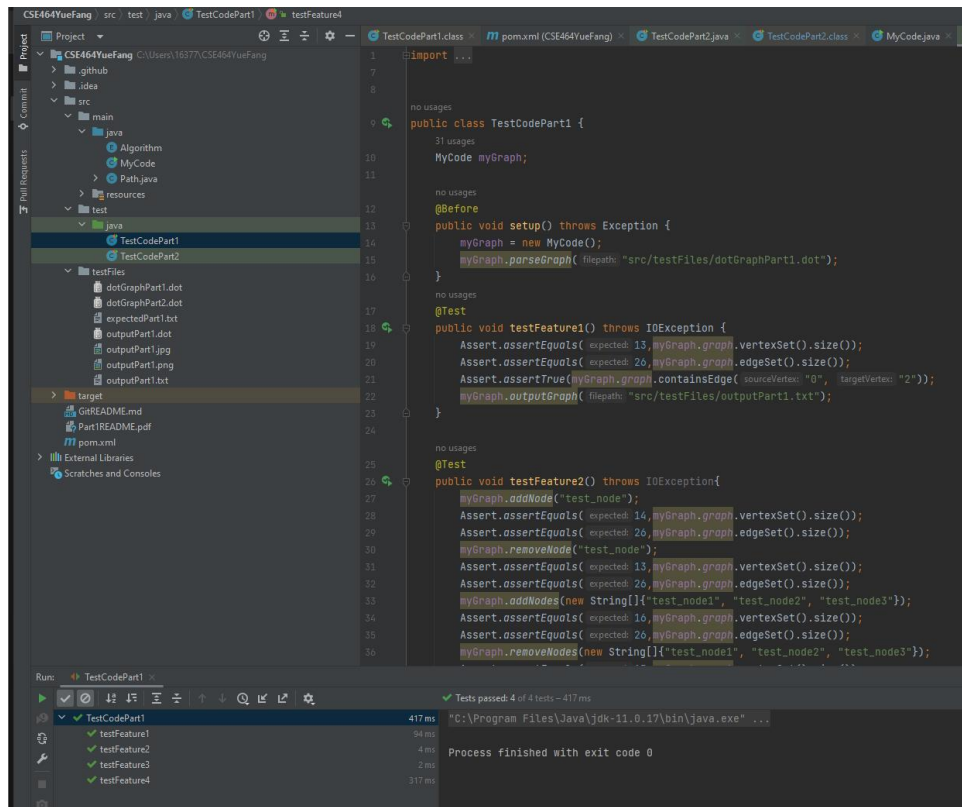
This project implements the following functions:

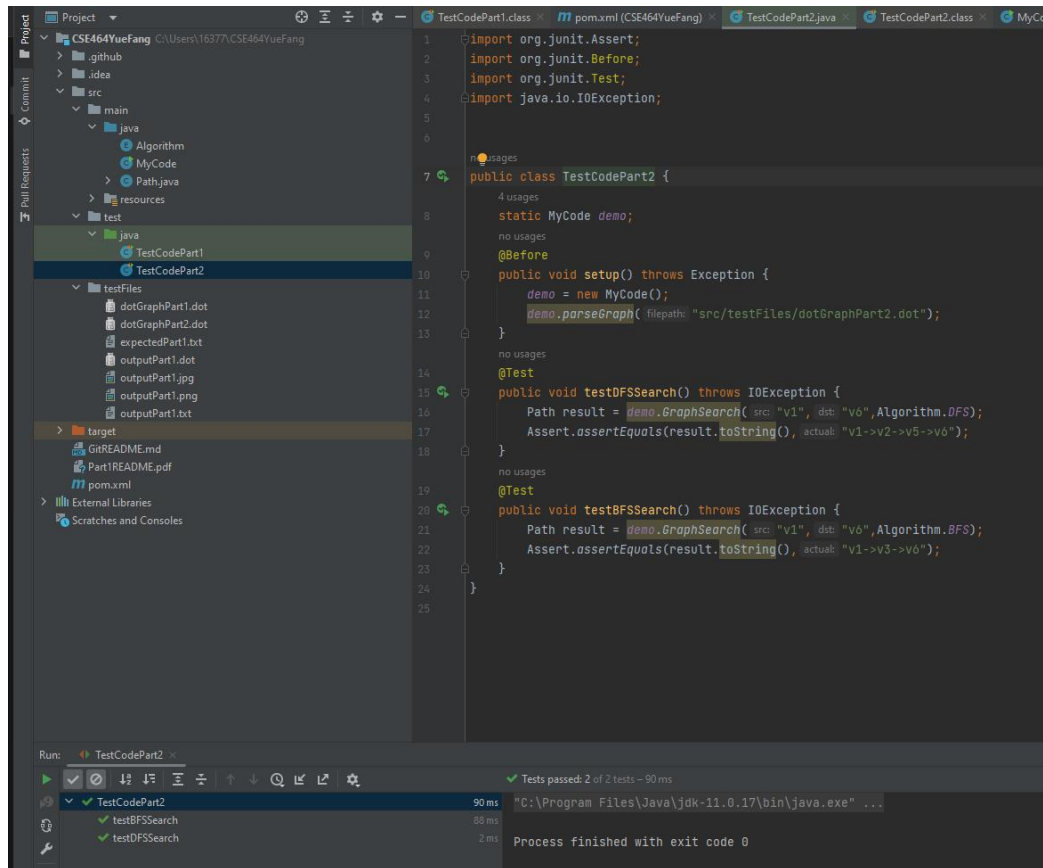
DFS Search : This function uses the DFS algorithm to find the path from the source point to the target point.

BFS Search : This function uses the BFS algorithm to find the path from the source point to the target point.

When merge the two functions, a conflict has arisen. To solve it, we add an enum parameter for the function. If the parameter == DFSTYPE, we use DFS algorithm to find the path from the source point to the target point, else, we use BFS algorithm to find the path from the source point to the target point.

I set two test cases. One is for part1, which is same as the case I use in part1. One is for part 2. This is a new case write for part2. Input in these test cases can be use as the example to test the code. If testCodePart1 got the last test cannot pass (I will try to make sure that will not happen, but there is still some chances for this problem happen. I think zip the folder cause this problem.), please use the exceptedPart1 in submission folder to replace the original one. The fail will be solved. This replace file is 100% same as the original, but use it will solve this problem.

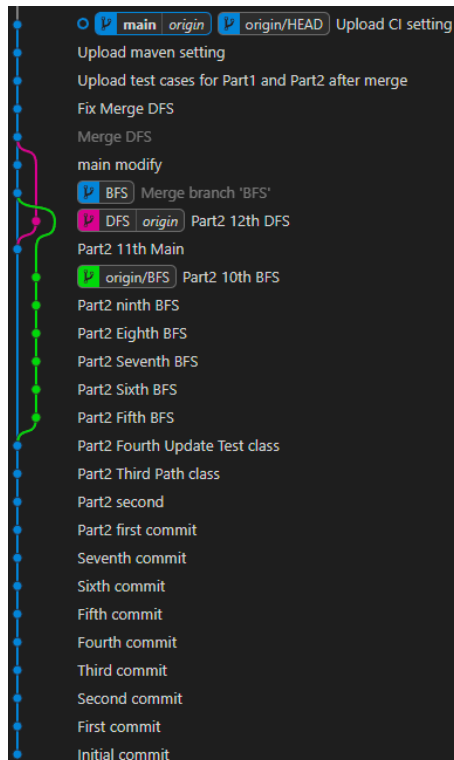




I got some problems when the first time I try to merge BFS and DFS, and I committed some useless code because of this, so I remove or modify these commits and the Workflows for them because these will influence my future work. If you find Workflow lost some runs, please do not feel strange. If this makes some problems for your view, sorry for that. Also, there are some problems with maven (forgot add path for test) and ci.yml (use wrong code, I should use "mvn test", but I use "mvn package"), so I upload a new version of them after I finish all the merges.

Following are the screen shot of my process:

Screen shot of my path graph:



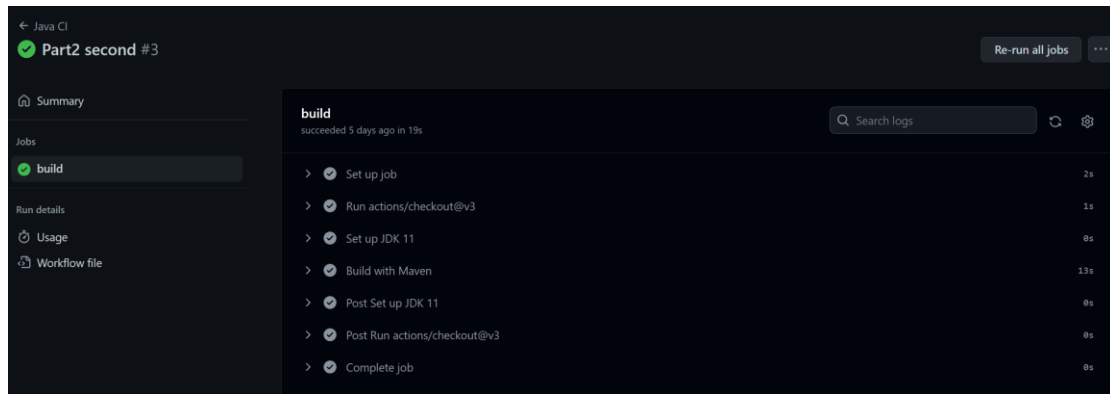
Screen shot for set maven support and CI: (final commit name: **Part2 second**)



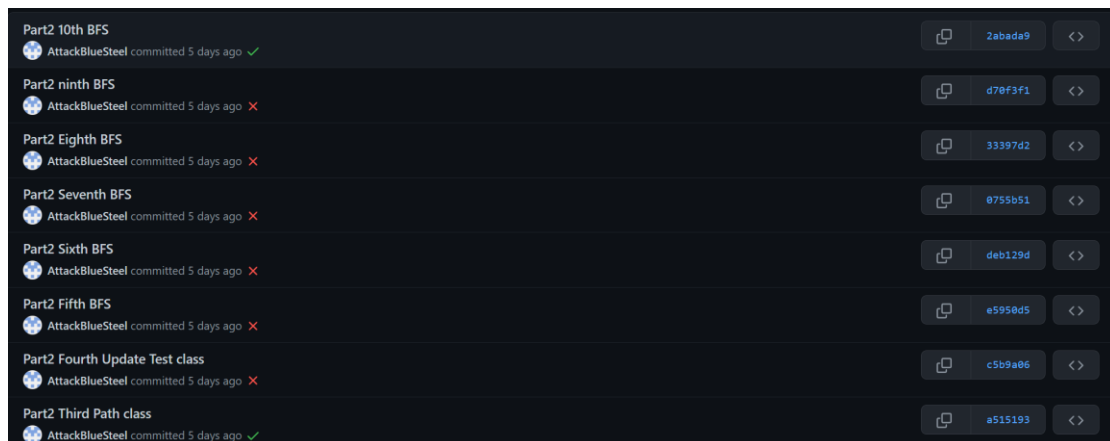
The first commit did not pass CI because of the path problem so I make some changes and do the second commit based on the first one. Second is the final result.

 AttackBlueSteel	Part2 second	✓ a54d3f2 5 days ago	🕒 10 commits
📁 .github/workflows	Part2 first commit		5 days ago
📁 .idea	Part2 second		5 days ago
📁 src	Part2 second		5 days ago
📁 target/classes	Part2 second		5 days ago
📄 README.md	Initial commit		last month
📄 README.pdf	Part2 second		5 days ago
📄 pom.xml	Part2 second		5 days ago

No change for files, only add maven and CI

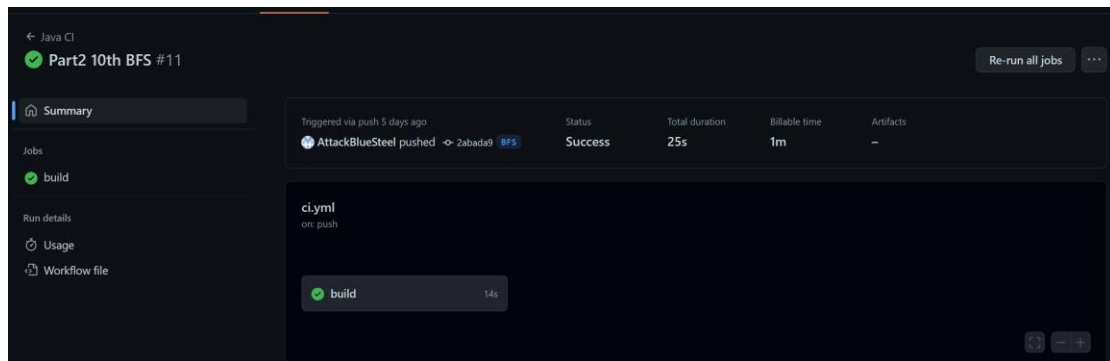


Screenshot for create BFS and make sure CI pass (Final commit name: Part2 10th BFS)



The change for this part is add the Path class, new test case, add some code, then create BFS branch and make sure the commit can pass CI. Because of some code problems, I changed some of my code and then make this part pass the CI. Part 2 10th BFS is the final result. **The command I use in my git to create new branch is `git branch BFS`, then I changed my code in local repo and use `git checkout BFS` to switch to BFS branch and do git commit and git push.**

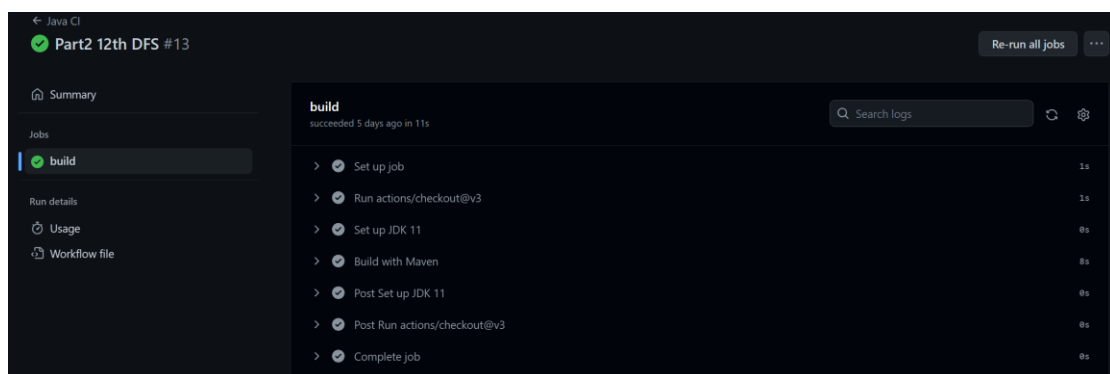
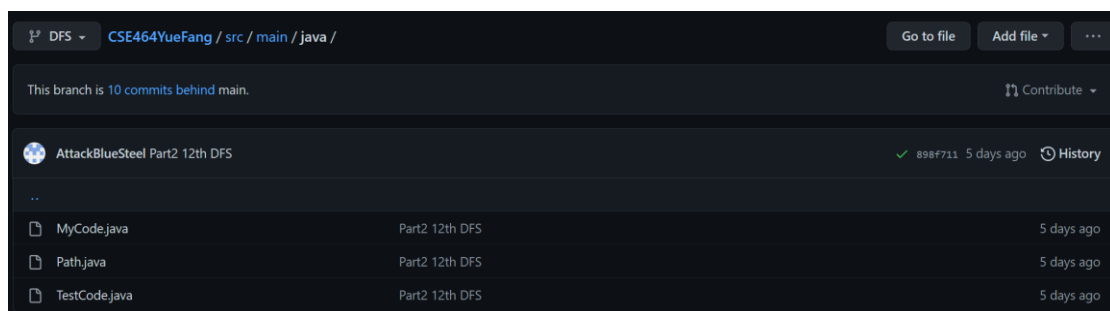




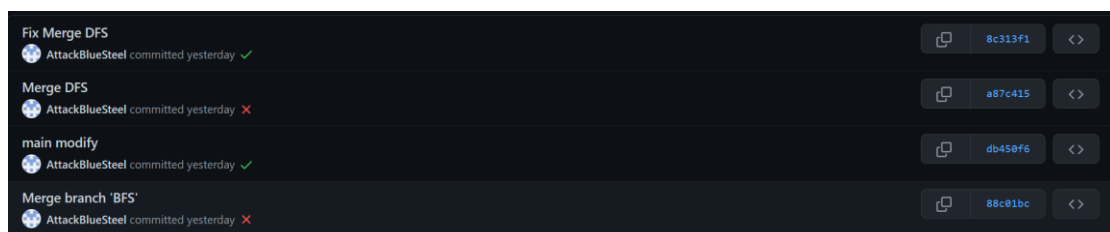
Screenshot for create DFS and make sure CI pass (Final commit name: Part2 12th BFS)



The change for this part is add the Path class, new test case, change some code, then create DFS branch and make sure the commit can pass CI, which is similar as the BFS part. Based on the experience on BFS, I only use one commit to finish this step. **The command I use in my git to create new branch is `git branch DFS`, then I changed my code in local repo and use `git checkout DFS` to switch to BFS branch and do git commit and git push.**



Screenshot for merge BFS, DFS and make sure CI pass (Final commit name: Fix Merge DFS)



The change for this part is merge BFS to the main, and then merge DFS to the main. **The command I use is `git checkout main` to switch to main, and then `git merge BFS` to merge BFS to main and do similar merge code to DFS. I merge BFS first, and there is no conflict.**

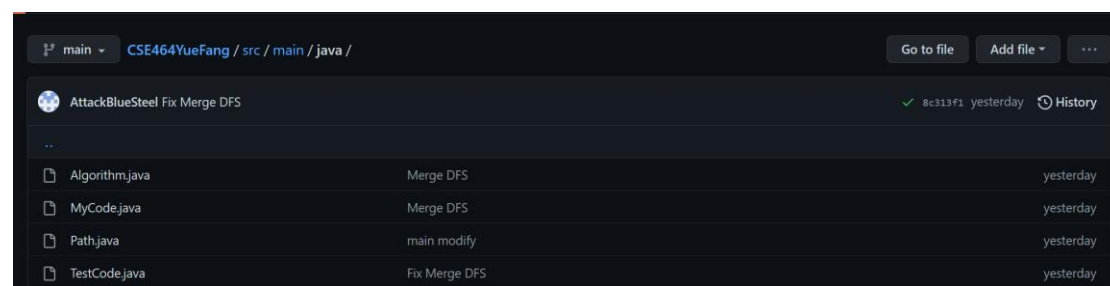
which is what I hope. But there is some problem with the Path class. Because of some actions I did in the past commit, GitHub think I may not want to keep the path class, so GitHub delete the path when I try to merge BFS. I do some changes and then make a new commit [main modify](#) and pass the CI after I merge BFS successfully. Then I merge DFS use similar way, but I got a conflict (see screen shot). So I open my code in VScode and accept both changes and write an enum Algorithm to make sure the program can switch the function. Then I do the commit, which is called Merge DFS. But I forgot to change some code in my test class, and this make this commit cannot pass CI. So I fix the problem and do another commit, which is called [Fix Merge DFS](#). Which should be the last commit for the coding part.

```
16377@DESKTOP-7B49EA7 MINGW64 ~/CSE464YueFang (DFS)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

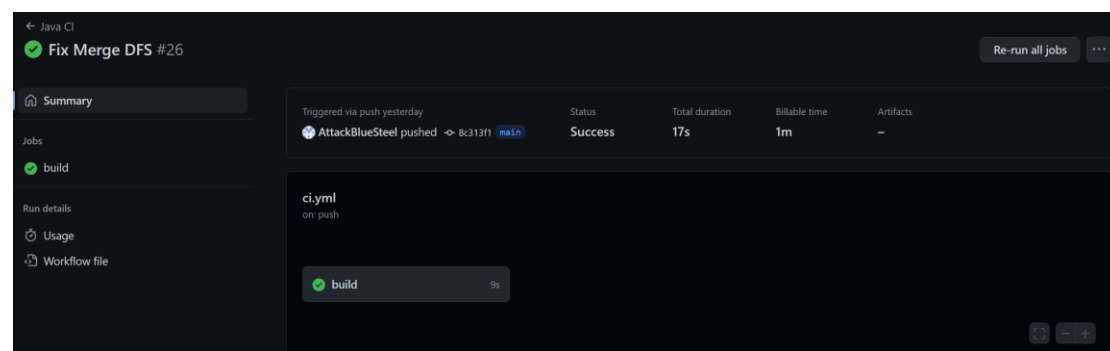
16377@DESKTOP-7B49EA7 MINGW64 ~/CSE464YueFang (main)
$ git merge BFS
Merge made by the 'ort' strategy.
 pom.xml | 5 +++++
 src/main/java/MyCode.java | 30 ++++++++++++++++++++++-----
 2 files changed, 33 insertions(+), 2 deletions(-)

16377@DESKTOP-7B49EA7 MINGW64 ~/CSE464YueFang (main)
$ git merge DFS
Auto-merging src/main/java/MyCode.java
CONFLICT (content): Merge conflict in src/main/java/MyCode.java
Automatic merge failed; fix conflicts and then commit the result.

16377@DESKTOP-7B49EA7 MINGW64 ~/CSE464YueFang (main|MERGING)
$
```



main - CSE464YueFang / src / main / java /		
AttackBlueSteel Fix Merge DFS ✓ 8c313f1 yesterday History		
Algorithm.java	Merge DFS	yesterday
MyCode.java	Merge DFS	yesterday
Path.java	main modify	yesterday
TestCode.java	Fix Merge DFS	yesterday



Java CI

Fix Merge DFS #26 Re-run all jobs

Summary

Jobs

- build

Run details

- Usage
- Workflow file

Triggered via push yesterday

AttackBlueSteel pushed · 8c313f1 main

Status: Success

Total duration: 17s

Billable time: 1m

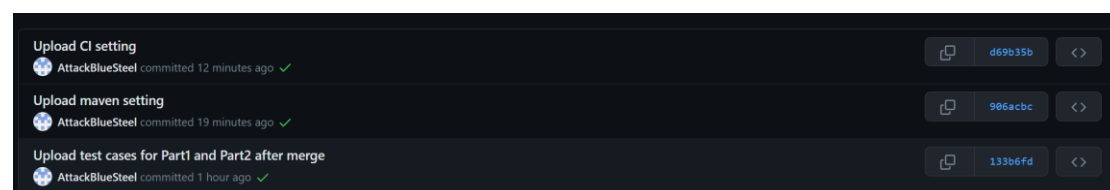
Artifacts: -

ci.yml

ort: push

build 0s

This is the final version of my work:



Upload CI setting

AttackBlueSteel committed 12 minutes ago ✓

Upload maven setting

AttackBlueSteel committed 19 minutes ago ✓

Upload test cases for Part1 and Part2 after merge

AttackBlueSteel committed 1 hour ago ✓

After I finish the merge, I found some error for my test cases ([I delet the test case for part1](#), [and use a new test case for part2](#), but seems like we need both, so I add the test case for [part1 back](#)), maven ([forgot to add test path](#)) and CI ([use wrong command, I should use mvn test](#), but I use mvn package). So I upload a new version of them to make sure the program

is good to run and easier to be viewed.

main 3 branches 0 tags Go to file Add file <> Code

Your main branch isn't protected
Protect this branch from force pushing or deletion, or require status checks before merging. [Learn more](#) Protect this branch

AttackBlueSteel Upload CI setting ✓ d69b35b 12 minutes ago 27 commits

File	Upload CI setting	Time
.github/workflows	Upload CI setting	12 minutes ago
.idea	Upload maven setting	19 minutes ago
src	Upload maven setting	19 minutes ago
target	Upload CI setting	12 minutes ago
GitREADME.md	Upload test cases for Part1 and Part2 after merge	1 hour ago
Part1README.pdf	Upload test cases for Part1 and Part2 after merge	1 hour ago
pom.xml	Upload maven setting	19 minutes ago

Java CI
Upload CI setting #29 Re-run all jobs

Summary

Jobs

- build

Run details

Usage

Workflow file

Triggered via push 16 minutes ago

AttackBlueSteel pushed · d69b35b main

Status Success

Total duration 34s

Billable time 1m

Artifacts -

ci.yml
on: push

build 23s

```
3-3.0.0.jar (12 kB at 351 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/surefire/common-junit4/3.0.0/common-junit
4-3.0.0.jar (26 kB at 455 kB/s)
[INFO]
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running TestCodePart1
[INFO] Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.354 s - in TestCodePart1
[INFO] Running TestCodePart2
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 s - in TestCodePart2
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 6, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] --- jar:3.3.0:jar (default-jar) @ CSE464YueFang ---
[INFO] Building jar: C:\Users\16377\CSE464YueFang\target\CSE464YueFang-1.0-SNAPSHOT.jar
[INFO] Copying org.example:CSE464YueFang:pom:1.0-SNAPSHOT to project local repository
[INFO] Copying org.example:CSE464YueFang:jar:1.0-SNAPSHOT to project local repository
[INFO] Copying org.example:CSE464YueFang:pom:consumer:1.0-SNAPSHOT to project local repository
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 2.577 s
[INFO] Finished at: 2023-03-28T20:56:15-07:00
[INFO]
[INFO] -----
C:\Users\16377\CSE464YueFang>
```

The code I submit on canvas will have three folders for three branches on GitHub, they are main, BFS, and DFS. Please choose the correct folder, thank you.