

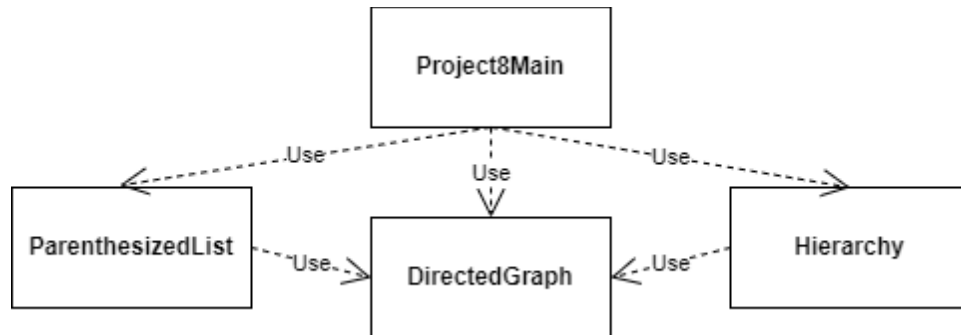
Project 4 Test Report

Chloe Ingram

09 May 2022

CMSC 350 - 7380

Part 1: UML Diagram



Here, the main project class uses all of the class documents. Additionally, the hierarchy and parenthesized list classes utilize the directed graph class, as that is where the DFSActions interface is defined.

Part 2: Test Results

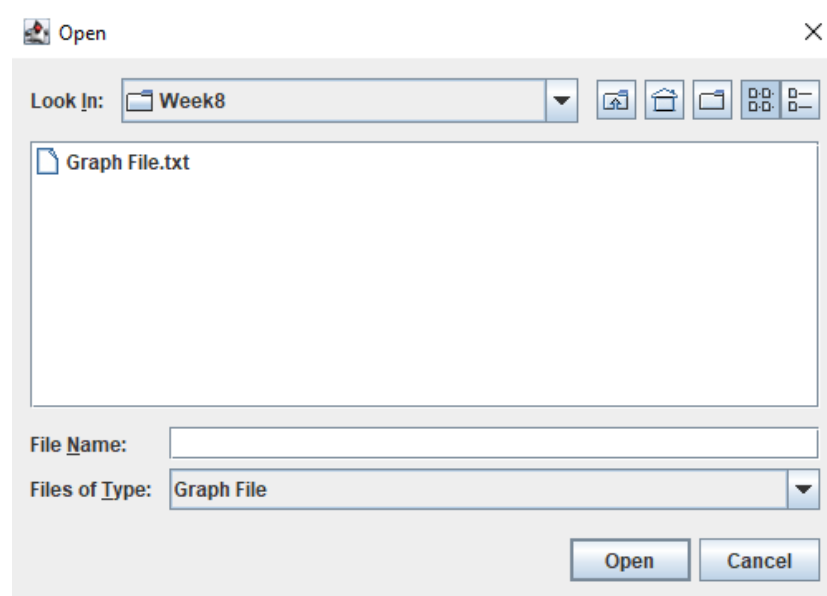


Figure 1. JFileChooser

```

<terminated> Project8Main [Java Application] C:\Users\pc\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_6

Unreachable vertices: ClassI
ClassA(ClassC * ClassE(ClassB(ClassD ClassG )ClassF ClassH )ClassJ(ClassB(ClassD ClassG )))

Unreachable vertices: ClassI
ClassA
  ClassC *
  ClassE
    ClassB
      ClassD
      ClassG
    ClassF
    ClassH
  ClassJ
    ClassB
      ClassD
      ClassG
Completed

```

Figure 2. Output from Project 4-provided Input

```

Graph File2 - Notepad
File Edit Format View Help
B C D F
F G
H I

```

Figure 3. Input 2 File

```

Console Problems Debug
<terminated> Project8Main [Java Applicatio

Unreachable vertices: H I
B(C D F(G ))

Unreachable vertices: H I
B
  C
  D
  F
    G
Completed

```

Figure 4. Input 2's Output

```
<terminated> Project8Main [Java Application] C:\Users\p... Graph Files - Notepad
File Edit Format
A B C
C B

Unreachable vertices:
A(B C(B ))

Unreachable vertices:
A
  B
  C
    B
Completed
```

Figure 5. Input 3 File and Output

```
Unreachable vertices:
A(B C(B )D * )

Unreachable vertices:
A
  B
  C
    B
    D *
Completed
```

```
Graph File4 - Notepad
File Edit Format View
A B C D
C B
D A
```

Figure 6. Input 4 File and Output

Test Table				
Test Case No.	Inputs	What is it testing?	Output	Pass/Fail
1	<i>Runs the code</i>	JFileChooser works correctly	JFileChooser worked correctly	Pass
2	ClassA ClassC ClassE ClassJ ClassB ClassD ClassG ClassC ClassA ClassE ClassB ClassF ClassH ClassJ ClassB ClassI ClassC	Testing against the project document-provided inputs to see if the code produces the same results	Figure 2	Pass
3	B C D F F G H I	Testing for unreachable vertices	Figure 4	Pass
4	A B C C B	Testing logic for circular dependencies as well as NO unreachable vertices	Figure 5	Pass
5	A B C D C B D A	Testing logic for circular dependencies	Figure 6	Pass

Part 3: Lessons Learned

This was the hardest project, but also the one I felt most confident with. I think I've finally figured out the alternative LinkedList approach, though the logic still gets hazy sometimes. I think I wished we used the library functions for ArrayList and LinkedList more, but I'm not mad about it, as during the test process, I created a lot of library-provided lists and tested those to ensure what I was doing was working!