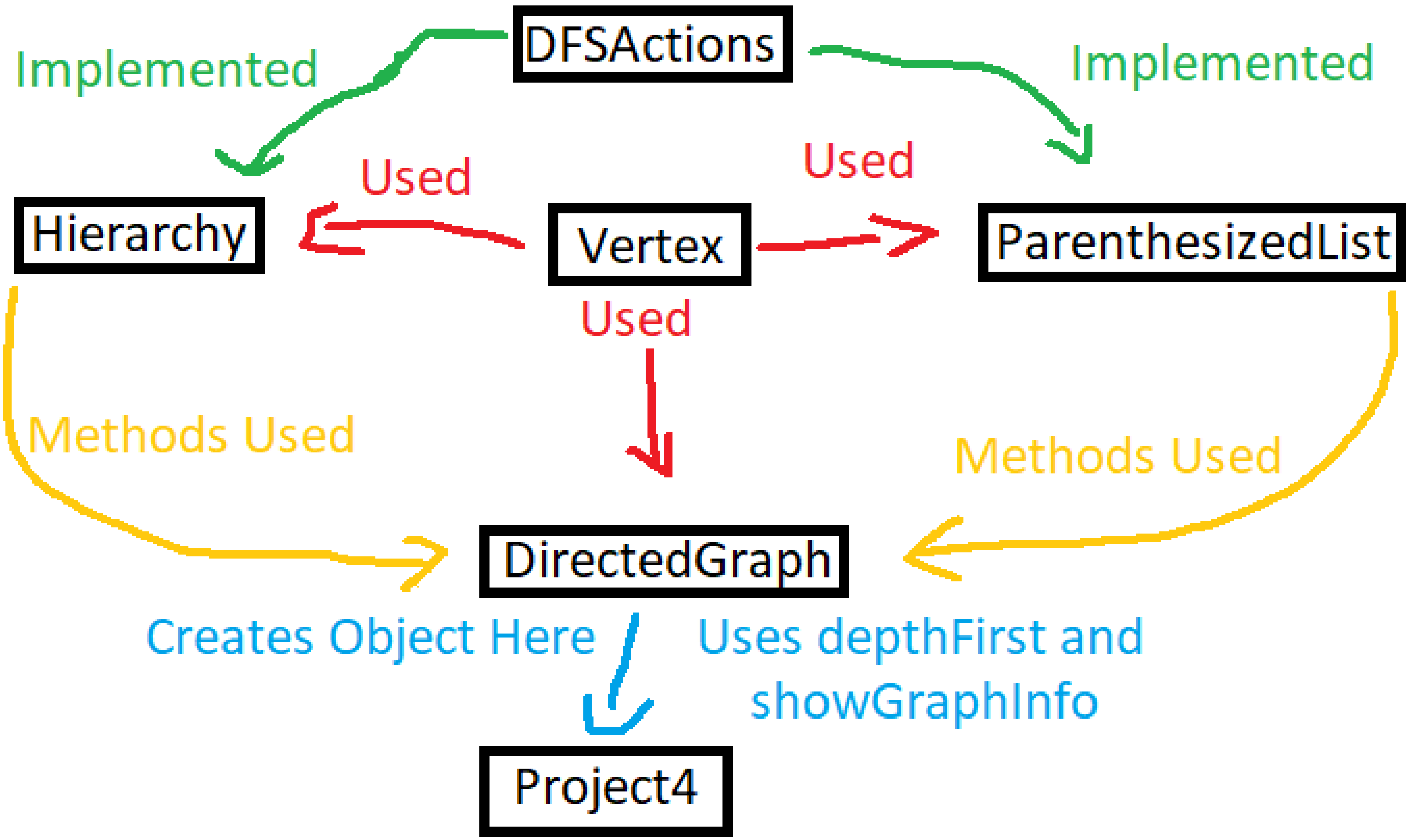
DFSActions
Implemented
Implemented
Used
Used
Hierarchy
Vertex
ParenthesizedList
Used
Methods Used
Methods Used
DirectedGraph
Creates Object Here
Uses depthFirst and showGraphInfo
Project4

Test Plan	Input	Output	Pass?

Test a graph with circular dependencies and an unreachable class.	(From file)
	ClassA ClassC ClassE ClassJ
	ClassB ClassD ClassG
	ClassC ClassA
	ClassE ClassB ClassF
	ClassH
	ClassJ ClassB
	ClassI ClassC
		Hierarchy:
		ClassA
		    ClassC *
		    ClassE
		        ClassB
		            ClassD
		            ClassG
		        ClassF
		        ClassH
		    ClassJ
		        ClassB
		            ClassD
		            ClassG
		
		Parenthesized:
		(ClassA (ClassC * ClassE (ClassB (ClassD ClassG) ClassF ClassH) ClassJ (ClassB (ClassD ClassG))))
		
		ClassI is unreachable.
		BUILD SUCCESSFUL (total time: 4 seconds)
			Yes

Test a graph with no circular dependencies and no unreachable classes.	(From file)
	ClassA ClassB ClassG
	ClassB ClassC ClassD
	ClassD ClassE ClassF
	ClassG ClassH ClassI
	ClassI ClassJ
		Hierarchy:
		ClassA
		    ClassB
		        ClassC
		        ClassD
		            ClassE
		            ClassF
		    ClassG
		        ClassH
		        ClassI
		            ClassJ
		
		Parenthesized:
		(ClassA (ClassB (ClassC ClassD (ClassE ClassF)) ClassG (ClassH ClassI (ClassJ))))
		
		All classes are reachable
		BUILD SUCCESSFUL (total time: 5 seconds)
			Yes

When I first started this project, I was quite overwhelmed. I couldn’t really wrap my head around as to how to go about creating a directed graph. I decided to just buckle down and try to figure it out. I watched plenty of videos, read plenty of articles, and frequently referenced the textbook. It too me many hours of trial and error but I eventually figured it out (at least I think so). The pseudocode was once again indispensable and helped me figure out how it all worked together. I had a rough time trying to properly implement the methods in Hierarchy and ParenthesizedList so I implemented code I found. I was able to trim and simplify the implemented code down, this helped me to ensure that I knew how it the code worked. I also found some areas in my code that were overly complicated so the implemented code helped simplify things too.