

Task 1:

Size:

1. 22539
2. HTMLEditor @ 2144 LOC
3. Every time there is something on a line, other than a return character, Metrics counts this as a LOC. This measure also includes solitary “}”s.

Cohesion:

1. LCOM2 is the ratio of methods that don't access a particular member variable to the total number of member variables. How it is calculated: $LCOM2(C) = 1 - e / (n * a)$; where C = class, e = edges, n = number of methods (within C) and a = number of member variables (attributes).
2. There are many classes with an LCOM2 of 0, which would indicate complete or perfect cohesion. This would mean that LCOM2 is either 0 or undefined, which is a good thing. This level of cohesion would indicate that either there is only one edge, one straight line of execution, no conditionals or loops, or either there are no member variables or methods.

Complexity:

1. 1.746 mean, 16 max
2. EventsManager @ 16 CC
3. In EventsManager line 230, method getEvent, there was a check for null on 'd', where null would be returned if this variable of type Day was null. I included this condition in the for loop, so if 'd' was null, it would not enter this for loop and skip to line 237, where null is returned. By moving this conditional, I reduced the number of edges, bringing our ratio, as discussed above, closer to 0. This resulted a CC reduction in main.java.memoranda from 1.746 to 1.743.

Package-level Coupling:

1. Afferent - The number of classes that are external the observed package in which it depends. “Incoming” dependencies.

Efferent – The number of classes from other packages that the classes in the observed package are dependent. “Outgoing” dependencies.

2. main.java.memoranda.util @ 57 afferent couplings
3. main.java.memoranda.ui @ 49 efferent couplings

Worst quality:

The strongest candidate for worst quality would be the `HTMLEditor` class. At a CC of 42, from `setImageProperties`, this is an immensely complex class due to this method's implementation, because of the poor structuring of the try-catch blocks. The same method requires 9 parameters to set image properties, which could be delegated to a Builder design pattern. Based on the CC alone, this is a risky class in its current implementation, and would likely be difficult to refactor.

Task 2:

1.

Metric	Total	Mean	Std. Dev	Maximur	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.24	2.851	42	/SER316-Spring-2018/src/main/java/memoran	<code>setTableProperties</code>
▶ Number of Parameters (avg/max per method)		0.928	1.097	9	/SER316-Spring-2018/src/main/java/memoran	<code>setImageProperties</code>
▶ Nested Block Depth (avg/max per method)		1.402	0.962	8	/SER316-Spring-2018/src/main/java/memoran	<code>getNotesForPeriod</code>
▶ Affarent Coupling (avg/max per packageFragment)		19.339	19.653	97	/SER316-Spring-2018/src/main/java/memoran	
▶ Effarent Coupling (avg/max per packageFragment)		11.848	15.276	48	/SER316-Spring-2018/src/main/java/memoran	
▶ Instability (avg/max per packageFragment)		0.36	0.247	0.778	/SER316-Spring-2018/src/main/java/memoran	
▶ Abstractness (avg/max per packageFragment)		0.111	0.137	0.333	/SER316-Spring-2018/src/main/java/memoran	
▶ Normalized Distance (avg/max per packageFragment)		0.529	0.237	1	/SER316-Spring-2018/src/main/java/memoran	
▶ Depth of Inheritance Tree (avg/max per type)		2.652	1.934	6	/SER316-Spring-2018/src/main/java/memoran	
▶ Weighted methods per Class (avg/max per type)	3253	14.143	25.513	242	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Children (avg/max per type)	60	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Overriden Methods (avg/max per type)	58	0.257	0.681	4	/SER316-Spring-2018/src/main/java/memoran	
▶ Lack of Cohesion of Methods (avg/max per type)		0.262	0.398	1.2	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Attributes (avg/max per type)	1326	5.765	14.118	101	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Static Attributes (avg/max per type)	136	0.591	1.793	12	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Methods (avg/max per type)	1268	5.517	6.813	42	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Static Methods (avg/max per type)	183	0.796	2.51	17	/SER316-Spring-2018/src/main/java/memoran	
▶ Specialisation Index (avg/max per type)		0.15	0.467	5	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Classes (avg/max per packageFragment)	230	25.556	29.833	92	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Interfaces (avg/max per packageFragment)	16	1.778	3.282	11	/SER316-Spring-2018/src/main/java/memoran	
▶ Number of Packages	9					
▶ Total Lines of Code	22569					
▶ Method Lines of Code (avg/max per method)	15667	10.79	28.233	346	/SER316-Spring-2018/src/main/java/memoran	<code>plot</code>

7. after:

Metric	Total	Mean	Std. Dev	Maximur	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.24	2.851	42	/SER316-Spring-2018/src/main/java/memoranda	setTableProperties
▶ Number of Parameters (avg/max per method)		0.928	1.097	9	/SER316-Spring-2018/src/main/java/memoranda	setImageProperties
▶ Nested Block Depth (avg/max per method)		1.402	0.962	8	/SER316-Spring-2018/src/main/java/memoranda	getNotesForPeriod
▶ Afferent Coupling (avg/max per packageFragment)		21.2	19.574	55	/SER316-Spring-2018/src/main/java/memoranda	
▶ Efferent Coupling (avg/max per packageFragment)		10.8	14.263	48	/SER316-Spring-2018/src/main/java/memoranda	
▶ Instability (avg/max per packageFragment)		0.338	0.242	0.778	/SER316-Spring-2018/src/main/java/memoranda	
▶ Abatementness (avg/max per packageFragment)		0.172	0.301	1	/SER316-Spring-2018/src/main/java/memoranda	
▶ Normalized Distance (avg/max per packageFragment)		0.52	0.25	1	/SER316-Spring-2018/src/main/java/memoranda	
▶ Depth of Inheritance Tree (avg/max per type)		2.652	1.894	6	/SER316-Spring-2018/src/main/java/memoranda	
▶ Weighted methods per Class (avg/max per type)	3253	14.143	25.533	242	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Children (avg/max per type)	80	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Overridden Methods (avg/max per type)	58	0.257	0.681	4	/SER316-Spring-2018/src/main/java/memoranda	
▶ Lack of Cohesion of Methods (avg/max per type)		0.262	0.388	1.2	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Attributes (avg/max per type)	1326	5.765	14.118	101	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Static Attributes (avg/max per type)	126	0.581	1.793	12	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Methods (avg/max per type)	1268	5.517	6.833	42	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Static Methods (avg/max per type)	183	0.796	2.51	17	/SER316-Spring-2018/src/main/java/memoranda	
▶ Specialization Index (avg/max per type)		0.15	0.487	5	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Classes (avg/max per packageFragment)	200	23	28.174	52	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Interfaces (avg/max per packageFragment)	16	1.6	3.165	11	/SER316-Spring-2018/src/main/java/memoranda	
▶ Number of Packages	10					
▶ Total Lines of Code	22536					
▶ Method Lines of Code (avg/max per method)	15467	10.79	28.223	346	/SER316-Spring-2018/src/main/java/memoranda	init

8. Afferent coupling increased slightly from 19.333 to 21.2, which is to be expected, since the interfaces that were in main.java.memoranda (main) are now in main.java.memoranda.interfaces (interfaces). This would increase the need for imports for classes, particularly in main, that utilize the newly relocated interfaces, and since they are of a new package, this would increase this particular metric.

Task 3:

1. Refactored:

Class: main.java.memoranda.ui.htmleditor.HTMLEditor.java

Code smell: Large Class

I moved many of the ActionListeners into their own external classes, since some of them are quite complicated, and removed all of the commented code. This class is enormous, and all the commented code did not help. Now the class is under 2000 lines, which is still large.

4.

Metric	Total	Mean	Std. Dev	Maximur	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.239	2.849	42	/SER316-Spring-2018/src/main/java/memorand	setTableProperties
▶ Number of Parameters (avg/max per method)		0.934	1.097	9	/SER316-Spring-2018/src/main/java/memorand	setImageProperties
▶ Nested Block Depth (avg/max per method)		1.402	0.962	8	/SER316-Spring-2018/src/main/java/memorand	getNotesForPeriod
▶ Affluent Coupling (avg/max per package/fragment)		21.7	19.422	55	/SER316-Spring-2018/src/main/java/memorand	
▶ Effluent Coupling (avg/max per package/fragment)		11.2	14.435	49	/SER316-Spring-2018/src/main/java/memorand	
▶ Instability (avg/max per package/fragment)		0.346	0.255	0.778	/SER316-Spring-2018/src/main/java/memorand	
▶ Abstractionness (avg/max per package/fragment)		0.172	0.301	1	/SER316-Spring-2018/src/main/java/memorand	
▶ Normalized Distance (avg/max per package/fragment)		0.512	0.258	1	/SER316-Spring-2018/src/main/java/memorand	
▶ Depth of Inheritance Tree (avg/max per type)		2.652	1.934	6	/SER316-Spring-2018/src/main/java/memorand	
▶ Weighted methods per Class (avg/max per type)	3255	14.152	25.529	242	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Children (avg/max per type)	60	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Overridden Methods (avg/max per type)	59	0.257	0.681	4	/SER316-Spring-2018/src/main/java/memorand	
▶ Lack of Cohesion of Methods (avg/max per type)		0.262	0.388	1.2	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Attributes (avg/max per type)	1236	5.809	14.102	101	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Static Attributes (avg/max per type)	136	0.591	1.793	12	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Methods (avg/max per type)	1271	5.526	6.829	42	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Static Methods (avg/max per type)	183	0.794	2.91	17	/SER316-Spring-2018/src/main/java/memorand	
▶ Specialization Index (avg/max per type)		0.149	0.484	5	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Classes (avg/max per package/fragment)	230	2.3	28.174	92	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Interfaces (avg/max per package/fragment)	16	1.6	2.169	11	/SER316-Spring-2018/src/main/java/memorand	
▶ Number of Packages	10					
▶ Total Lines of Code	22593					

Most notably, the mean CC decreased from 2.24 to 2.239, because the number of classes increased (ActionListeners), and most of those classes had a CC of 1 or 0. This seems like an artificial improvement, if anything.