Assignment 7

Michael Falgien

SER316-SpringB2018

Task 1

Size

1. What is Total Lines of Code (LOC) in the project? 22539
2. What is the largest single code file in the project and its Total LOC? The largest file is main.java.memoranda.ui.htmleditor.HTMLEditor.java which is 2,144 lines of code.
3. The method the Metrics tool uses to determine total LOC is physical LOC. I can tell that it uses Physical LOC and not Logical LOC is that it counts opening and closing brackets as a line of code. I manually counted the lines with any code, including brackets and excluding comments, and it was 28 LOC which is also what Metrics reports.

Cohesion

1. The Henderson-Sellers Lack of Cohesion method is defined as:

M = the set of methods defined by the class

F = the set of fields defined by the class

P(*f*) = the number of methods that access field *f*, where *f* is a member of F

<p> = mean of P(*f*) over F

LCOM2 = $\frac{<p> - |M|}{1-|M|}$

2. The class MergeSort in the package treetable has a score of 0 like many other classes. The cohesion is low due to the fact that there are few member variables in this class and it is an abstract class. Its cohesion score is also low because it is highly reusable. Because its member variables are of the class Object, this code can be reused by any class that extends Object.

Complexity

1. What is the cyclomatic complexity in the main package? The mean is 2.241 and the standard deviation is 2.851.
2. On average, the ParaBreakAction class in the HTMLEditor.java file and the CircleRegionContainment class in AltHTMLWriter.java have the worst CC at 7.5.

3.

Package-level Coupling

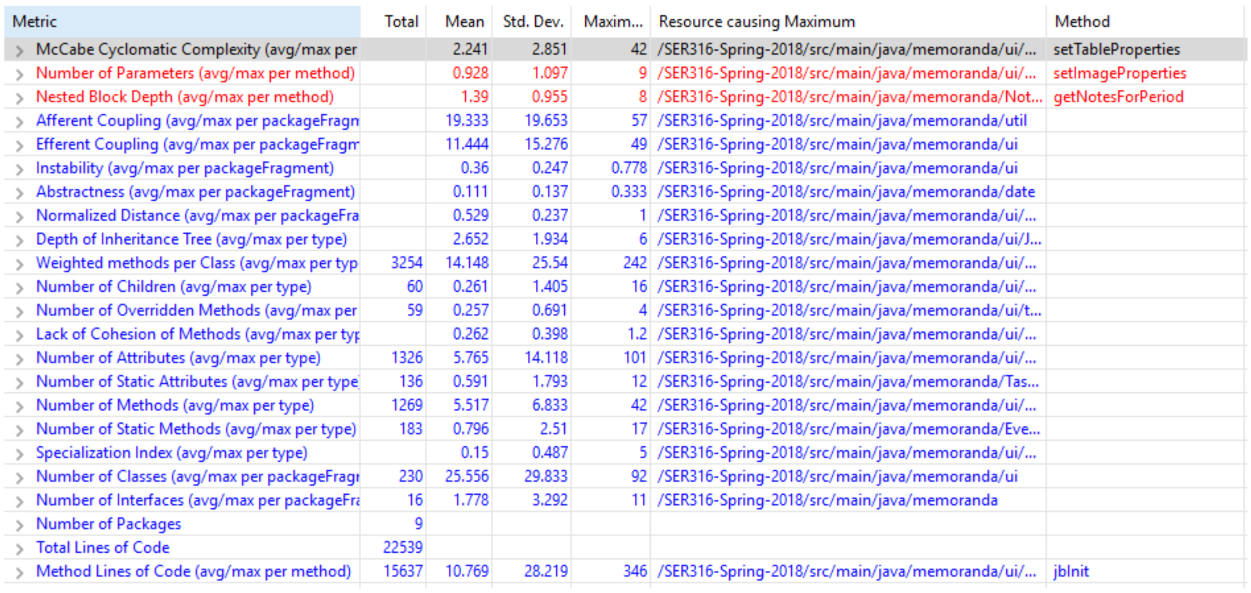
1. Afferent Coupling: The number of classes in other packages that depend upon classes within the package.
   Efferent Coupling: The number of classes in other packages that the classes in a package depend upon.

   Difference: Efferent coupling is the dependence on externalities whereas afferent coupling is internal dependencies. Also afferent is incoming dependencies and efferent is outgoing dependencies.
2. Worst Afferent Coupling Measure: main.java.memoranda.util with 57
3. Worst Efferent Coupling Measure: main.java.memoranda.ui with 49

Worst Quality

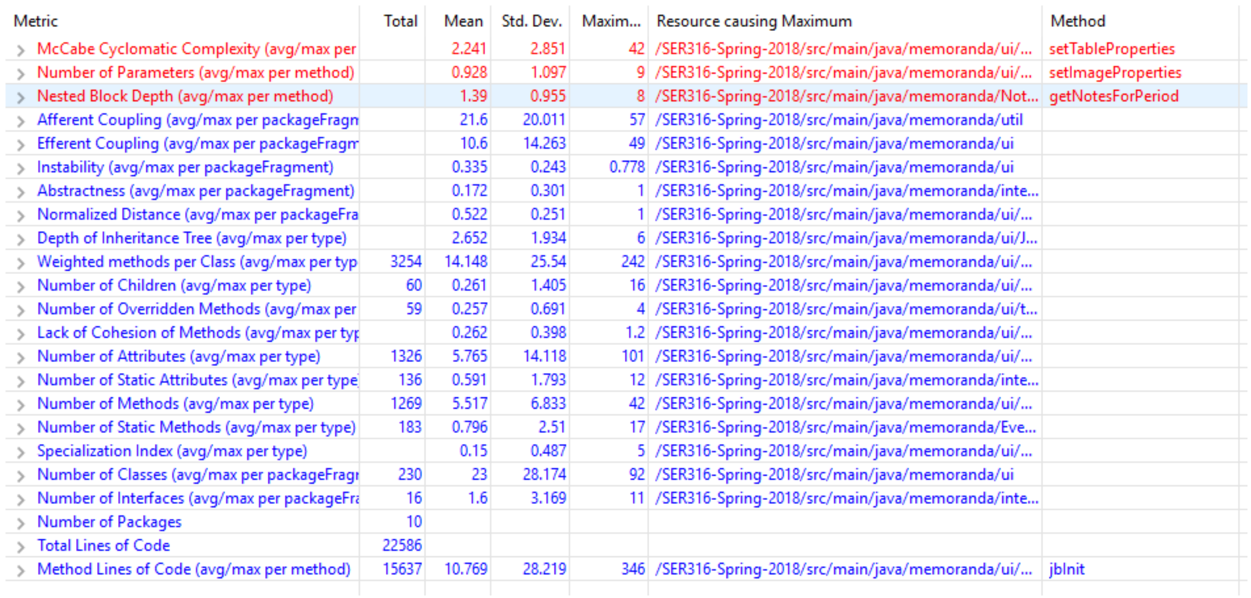
I think the worst quality class is HTMLEditor. This class has 101 attributes when the average is 5.8. The complexity is very high at 3.562 and also has the most lines of code. This class would be incredibly tough to debug given the large number of attributes, size in LOC and complexity.

Task 2



Metric	Total	Mean	Std. Dev.	Maxim...	Resource causing Maximum	Method
McCabe Cyclomatic Complexity (avg/max per		2.241	2.851	42	/SER316-Spring-2018/src/main/java/memoranda/ui/...	setTableProperties
Number of Parameters (avg/max per method)		0.928	1.097	9	/SER316-Spring-2018/src/main/java/memoranda/ui/...	setImageProperties
Nested Block Depth (avg/max per method)		1.39	0.955	8	/SER316-Spring-2018/src/main/java/memoranda/Not...	getNotesForPeriod
Afferent Coupling (avg/max per packageFragn		19.333	19.653	57	/SER316-Spring-2018/src/main/java/memoranda/util	
Efferent Coupling (avg/max per packageFragm		11.444	15.276	49	/SER316-Spring-2018/src/main/java/memoranda/ui	
Instability (avg/max per packageFragment)		0.36	0.247	0.778	/SER316-Spring-2018/src/main/java/memoranda/ui	
Abstractness (avg/max per packageFragment)		0.111	0.137	0.333	/SER316-Spring-2018/src/main/java/memoranda/date	
Normalized Distance (avg/max per packageFra		0.529	0.237	1	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Depth of Inheritance Tree (avg/max per type)		2.652	1.934	6	/SER316-Spring-2018/src/main/java/memoranda/ui/J...	
Weighted methods per Class (avg/max per typ	3254	14.148	25.54	242	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Children (avg/max per type)	60	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Overridden Methods (avg/max per	59	0.257	0.691	4	/SER316-Spring-2018/src/main/java/memoranda/ui/t...	
Lack of Cohesion of Methods (avg/max per typ		0.262	0.398	1.2	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Attributes (avg/max per type)	1326	5.765	14.118	101	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Static Attributes (avg/max per type	136	0.591	1.793	12	/SER316-Spring-2018/src/main/java/memoranda/Tas...	
Number of Methods (avg/max per type)	1269	5.517	6.833	42	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Static Methods (avg/max per type)	183	0.796	2.51	17	/SER316-Spring-2018/src/main/java/memoranda/Eve...	
Specialization Index (avg/max per type)		0.15	0.487	5	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Classes (avg/max per packageFragi	230	25.556	29.833	92	/SER316-Spring-2018/src/main/java/memoranda/ui	
Number of Interfaces (avg/max per packageFra	16	1.778	3.292	11	/SER316-Spring-2018/src/main/java/memoranda	
Number of Packages	9					
Total Lines of Code	22539					
Method Lines of Code (avg/max per method)	15637	10.769	28.219	346	/SER316-Spring-2018/src/main/java/memoranda/ui/...	jbInit

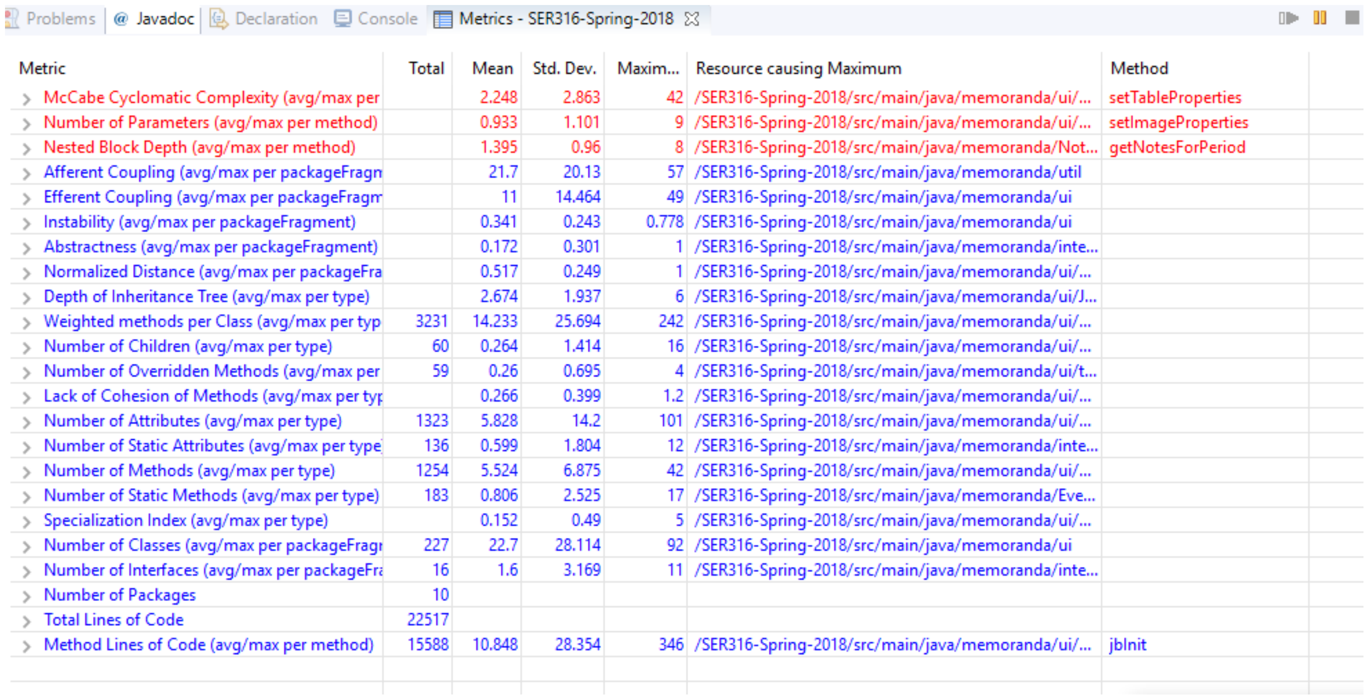
After



Metric	Total	Mean	Std. Dev.	Maxim...	Resource causing Maximum	Method
McCabe Cyclomatic Complexity (avg/max per		2.241	2.851	42	/SER316-Spring-2018/src/main/java/memoranda/ui/...	setTableProperties
Number of Parameters (avg/max per method)		0.928	1.097	9	/SER316-Spring-2018/src/main/java/memoranda/ui/...	setImageProperties
Nested Block Depth (avg/max per method)		1.39	0.955	8	/SER316-Spring-2018/src/main/java/memoranda/Not...	getNotesForPeriod
Afferent Coupling (avg/max per packageFragn		21.6	20.011	57	/SER316-Spring-2018/src/main/java/memoranda/util	
Efferent Coupling (avg/max per packageFragm		10.6	14.263	49	/SER316-Spring-2018/src/main/java/memoranda/ui	
Instability (avg/max per packageFragment)		0.335	0.243	0.778	/SER316-Spring-2018/src/main/java/memoranda/ui	
Abstractness (avg/max per packageFragment)		0.172	0.301	1	/SER316-Spring-2018/src/main/java/memoranda/inte...	
Normalized Distance (avg/max per packageFra		0.522	0.251	1	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Depth of Inheritance Tree (avg/max per type)		2.652	1.934	6	/SER316-Spring-2018/src/main/java/memoranda/ui/J...	
Weighted methods per Class (avg/max per typ	3254	14.148	25.54	242	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Children (avg/max per type)	60	0.261	1.405	16	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Overridden Methods (avg/max per	59	0.257	0.691	4	/SER316-Spring-2018/src/main/java/memoranda/ui/t...	
Lack of Cohesion of Methods (avg/max per typ		0.262	0.398	1.2	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Attributes (avg/max per type)	1326	5.765	14.118	101	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Static Attributes (avg/max per type	136	0.591	1.793	12	/SER316-Spring-2018/src/main/java/memoranda/inte...	
Number of Methods (avg/max per type)	1269	5.517	6.833	42	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Static Methods (avg/max per type)	183	0.796	2.51	17	/SER316-Spring-2018/src/main/java/memoranda/Eve...	
Specialization Index (avg/max per type)		0.15	0.487	5	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Classes (avg/max per packageFragi	230	23	28.174	92	/SER316-Spring-2018/src/main/java/memoranda/ui	
Number of Interfaces (avg/max per packageFra	16	1.6	3.169	11	/SER316-Spring-2018/src/main/java/memoranda/inte...	
Number of Packages	10					
Total Lines of Code	22586					
Method Lines of Code (avg/max per method)	15637	10.769	28.219	346	/SER316-Spring-2018/src/main/java/memoranda/ui/...	jbInit

A pretty easy change to notice after refactoring is the total LOC. This is caused by new import statements in classes that previously were contained in the same package as the interfaces. If a class uses the interface, after refactoring it is necessary to add an import statement which will lead to the increase in LOC. Along these lines, the total number of packages also increased. One of the metrics that changed is Afferent Coupling which got worse, but Efferent Coupling got better. The afferent coupling got worse due to the fact that it now depends upon a class that is in a different package, whereas before the refactoring, the dependencies were in the same package.

Task 3

1. Class: NoteListImpl in main.java.memoranda . In this class I made the private class NoteElement into its own class to reduce the size of the class. It is over 400 LOC which is pretty large for this project. I also made it into its own class so that it could be used later on in Task 2.2.
2. Duplicate code in NoteListImpl.java and EventsManager.java. Classes Year, Month and Day are now their own classes to reduce duplication.
3. 

Problems | Javadoc | Declaration | Console | Metrics - SER316-Spring-2018

Metric	Total	Mean	Std. Dev.	Maxim...	Resource causing Maximum	Method
McCabe Cyclomatic Complexity (avg/max per		2.248	2.863	42	/SER316-Spring-2018/src/main/java/memoranda/ui/...	setTableProperties
Number of Parameters (avg/max per method)		0.933	1.101	9	/SER316-Spring-2018/src/main/java/memoranda/ui/...	setImageProperties
Nested Block Depth (avg/max per method)		1.395	0.96	8	/SER316-Spring-2018/src/main/java/memoranda/Not...	getNotesForPeriod
Afferent Coupling (avg/max per packageFragn		21.7	20.13	57	/SER316-Spring-2018/src/main/java/memoranda/util	
Efferent Coupling (avg/max per packageFragm		11	14.464	49	/SER316-Spring-2018/src/main/java/memoranda/ui	
Instability (avg/max per packageFragment)		0.341	0.243	0.778	/SER316-Spring-2018/src/main/java/memoranda/ui	
Abstractness (avg/max per packageFragment)		0.172	0.301	1	/SER316-Spring-2018/src/main/java/memoranda/inte...	
Normalized Distance (avg/max per packageFra		0.517	0.249	1	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Depth of Inheritance Tree (avg/max per type)		2.674	1.937	6	/SER316-Spring-2018/src/main/java/memoranda/ui/J...	
Weighted methods per Class (avg/max per typ	3231	14.233	25.694	242	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Children (avg/max per type)	60	0.264	1.414	16	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Overridden Methods (avg/max per	59	0.26	0.695	4	/SER316-Spring-2018/src/main/java/memoranda/ui/t...	
Lack of Cohesion of Methods (avg/max per typ		0.266	0.399	1.2	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Attributes (avg/max per type)	1323	5.828	14.2	101	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Static Attributes (avg/max per type	136	0.599	1.804	12	/SER316-Spring-2018/src/main/java/memoranda/inte...	
Number of Methods (avg/max per type)	1254	5.524	6.875	42	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Static Methods (avg/max per type)	183	0.806	2.525	17	/SER316-Spring-2018/src/main/java/memoranda/Eve...	
Specialization Index (avg/max per type)		0.152	0.49	5	/SER316-Spring-2018/src/main/java/memoranda/ui/...	
Number of Classes (avg/max per packageFragı	227	22.7	28.114	92	/SER316-Spring-2018/src/main/java/memoranda/ui	
Number of Interfaces (avg/max per packageFra	16	1.6	3.169	11	/SER316-Spring-2018/src/main/java/memoranda/inte...	
Number of Packages	10					
Total Lines of Code	22517					
Method Lines of Code (avg/max per method)	15588	10.848	28.354	346	/SER316-Spring-2018/src/main/java/memoranda/ui/...	jbInit

4. Many of the metrics remained the same, but there are a few metrics that changed by small amounts. The afferent coupling measure did get smaller due to their being more dependencies contained within the package. Also LOC did reduce by about 70 lines of code. This is the size of all the new classes created and because they were each being used in a different location, we are now not reduplicating the code in separate files.

Describe the code smell: In task 1, the code smell was large class. I removed a smaller class contained in a larger class to make it into its own class. This would also be necessary for task 2. In task 2, I noticed that there was duplicate code between two classes. To fix this, I made a class for Year, Month and Day so that the code would not have to be reduplicated for each class.