Ethan Underhill
SER316 – Assignment 7

Task 1

Size

1. Total LOC main.java.memoranda: 2187
2. Largest single code file and LOC: EventsManager.java, 329 LOC
3. CurrentNote.java LOC method: Total LOC is 28, which is a simplistic physical LOC measure of every non-comment, non-whitespace line in the file (including closing brackets alone on a line, etc.). The Metrics tool also provides a Method Lines of Code measure.

Cohesion

1. Definition of LCOM2 and calculation method:
   Henderson-Sellers defines LCOM as $\frac{<r> - |M|}{1 - |M|}$, where **M** is the set of methods defined by the class, **F** is the set of fields defined by the class, *r*(*f*) is the number of methods that access field *f* (*f* is a member of F), and <*r*> is the mean of *r*(*f*) over **F**.
2. Highest cohesion class: Start.java (and many other classes in the package) is perfectly cohesive, as it has a LCOM score of 0.

Complexity

1. Cyclomatic complexity of main package: 16
2. Class with worst average McCabe Cyclomatic Complexity: EventsManager.java, 16
3. Reduced Cyclomatic Complexity in ProjectImpl.java getStatus method by one (from 6 to 5) by elevating code inside "if (prEnd == null)" condition to its own method.

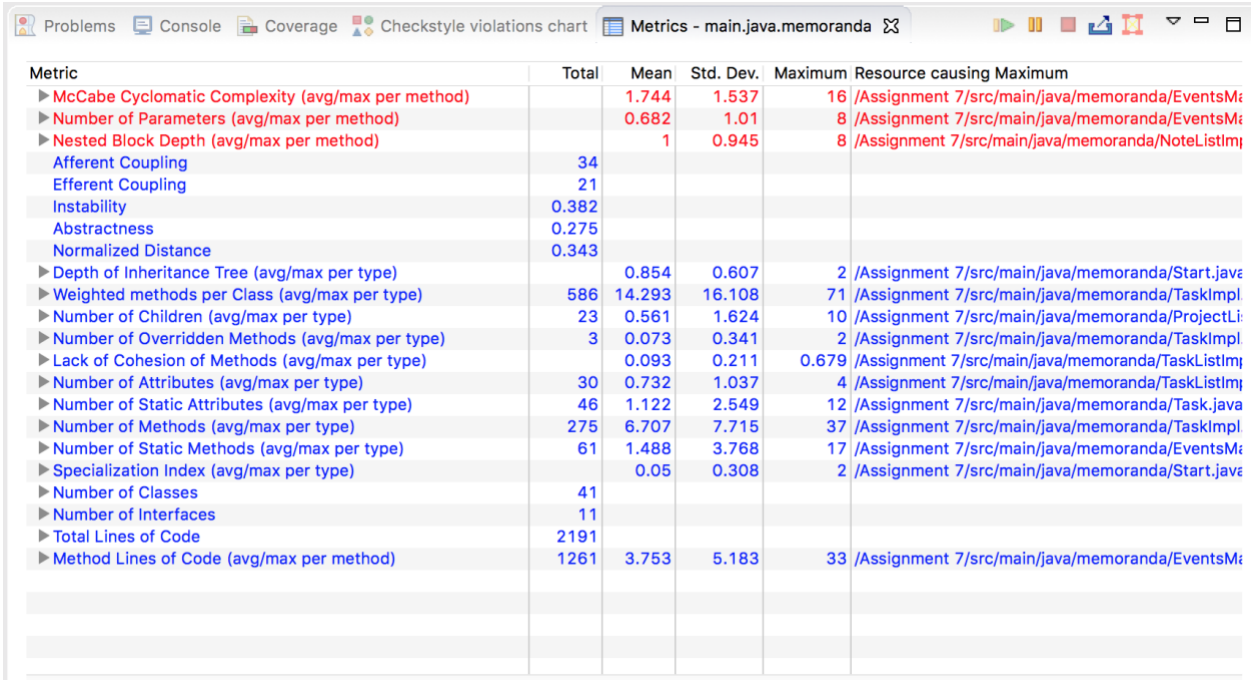
Package-Level Coupling

1. Afferent coupling is the number of classes in other packages that depend upon classes within the package (measure of package responsibility). Efferent coupling is the number of classes that classes in the package depend upon (measure of package dependence).
2. main.java.memoranda.util has the worst afferent coupling (57).
3. main.java.memoranda.ui has the worst efferent coupling (49).

Worst Quality

EventsManager.java is the worst in the package, based on Cyclomatic Complexity. The getRepeatableEventsForDate method has a CC value of 16, with a long and difficult-to-read method composed of many nested if-else statements. This then both has high overhead and is difficult to maintain or modify.

Task 2



Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum
McCabe Cyclomatic Complexity (avg/max per method)		1.744	1.537	16	/Assignment 7/src/main/java/memoranda/EventsMa
Number of Parameters (avg/max per method)		0.682	1.01	8	/Assignment 7/src/main/java/memoranda/EventsMa
Nested Block Depth (avg/max per method)		1	0.945	8	/Assignment 7/src/main/java/memoranda/NoteListImp
Afferent Coupling	34				
Efferent Coupling	21				
Instability	0.382				
Abstractness	0.275				
Normalized Distance	0.343				
Depth of Inheritance Tree (avg/max per type)		0.854	0.607	2	/Assignment 7/src/main/java/memoranda/Start.java
Weighted methods per Class (avg/max per type)	586	14.293	16.108	71	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Children (avg/max per type)	23	0.561	1.624	10	/Assignment 7/src/main/java/memoranda/ProjectLi
Number of Overridden Methods (avg/max per type)	3	0.073	0.341	2	/Assignment 7/src/main/java/memoranda/TaskImpl
Lack of Cohesion of Methods (avg/max per type)		0.093	0.211	0.679	/Assignment 7/src/main/java/memoranda/TaskListImp
Number of Attributes (avg/max per type)	30	0.732	1.037	4	/Assignment 7/src/main/java/memoranda/TaskListImp
Number of Static Attributes (avg/max per type)	46	1.122	2.549	12	/Assignment 7/src/main/java/memoranda/Task.java
Number of Methods (avg/max per type)	275	6.707	7.715	37	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Static Methods (avg/max per type)	61	1.488	3.768	17	/Assignment 7/src/main/java/memoranda/EventsMa
Specialization Index (avg/max per type)		0.05	0.308	2	/Assignment 7/src/main/java/memoranda/Start.java
Number of Classes	41				
Number of Interfaces	11				
Total Lines of Code	2191				
Method Lines of Code (avg/max per method)	1261	3.753	5.183	33	/Assignment 7/src/main/java/memoranda/EventsMa

Figure 1 Before

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum
McCabe Cyclomatic Complexity (avg/max per method)		2.029	1.725	16	/Assignment 7/src/main/java/memoranda/EventsMa
Number of Parameters (avg/max per method)		0.716	1.017	8	/Assignment 7/src/main/java/memoranda/EventsMa
Nested Block Depth (avg/max per method)		1.383	0.84	8	/Assignment 7/src/main/java/memoranda/NoteListImp
Afferent Coupling	31				
Efferent Coupling	16				
Instability	0.34				
Abstractness	0				
Normalized Distance	0.66				
Depth of Inheritance Tree (avg/max per type)		1.167	0.373	2	/Assignment 7/src/main/java/memoranda/Start.java
Weighted methods per Class (avg/max per type)	493	16.433	17.88	71	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Children (avg/max per type)	0	0	0	0	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Overridden Methods (avg/max per type)	3	0.1	0.396	2	/Assignment 7/src/main/java/memoranda/TaskImpl
Lack of Cohesion of Methods (avg/max per type)		0.127	0.237	0.679	/Assignment 7/src/main/java/memoranda/TaskListImp
Number of Attributes (avg/max per type)	30	1	1.095	4	/Assignment 7/src/main/java/memoranda/TaskListImp
Number of Static Attributes (avg/max per type)	29	0.967	2.008	7	/Assignment 7/src/main/java/memoranda/History.ja
Number of Methods (avg/max per type)	182	6.067	7.874	37	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Static Methods (avg/max per type)	61	2.033	4.278	17	/Assignment 7/src/main/java/memoranda/EventsMa
Specialization Index (avg/max per type)		0.068	0.359	2	/Assignment 7/src/main/java/memoranda/Start.java
Number of Classes	30				
Number of Interfaces	0				
Total Lines of Code	2068				
Method Lines of Code (avg/max per method)	1261	5.189	5.449	33	/Assignment 7/src/main/java/memoranda/EventsMa

Figure 2 After

8. Moving the interface classes into their own package reduced both afferent and efferent coupling (and therefore Instability, which is calculated from those metrics). Having the interfaces in their own package meant that other packages requiring only the interfaces did not have to be coupled with main.java.memoranda.

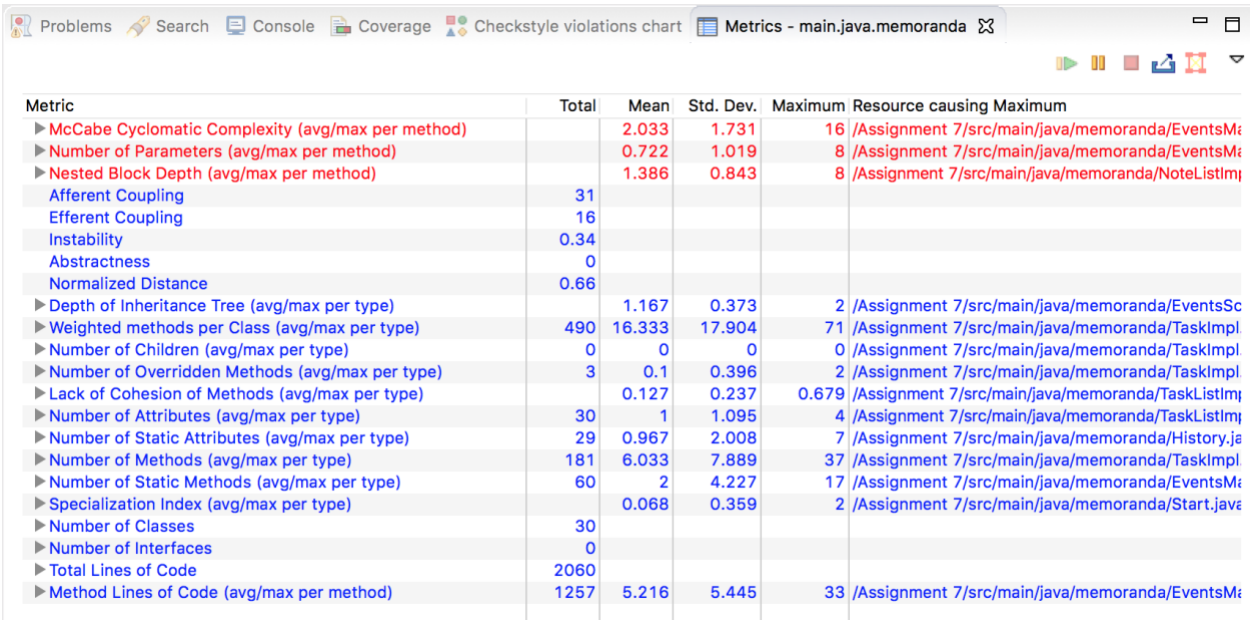
Task 3

2-1. CS: Too short identifiers. Changed method "**public** CalendarDate getEarliestStartDateFromSubTasks(ITask task)" in TaskListImpl.java (inside main.java.memoranda). This code was difficult to read due to identifiers like "d" and "dd" being

used in the method, so these were expanded to have more readable meaning (i.e. “date” and “earlyDate” replacing “d” and “dd” respectively). This is common throughout the Memoranda code and is very confusing in some of the longer methods.

2-2. CS: Middleman. I did not successfully change the code, instead marking this CS in IEventNotificationListener.java (in main.java.memoranda.interfaces) and DefaultEventNotifier.java and EventsScheduler.java (in main.java.memoranda). The interface effectively only has a single method, as the eventsChanged() method is empty in the DefaultEventNotifier class as well. I believe that the code could be refactored to remove DefaultEventNotifier and still function properly.

2-3.



Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum
McCabe Cyclomatic Complexity (avg/max per method)		2.033	1.731	16	/Assignment 7/src/main/java/memoranda/EventsMa
Number of Parameters (avg/max per method)		0.722	1.019	8	/Assignment 7/src/main/java/memoranda/EventsMa
Nested Block Depth (avg/max per method)		1.386	0.843	8	/Assignment 7/src/main/java/memoranda/NoteListIm
Afferent Coupling	31				
Efferent Coupling	16				
Instability	0.34				
Abstractness	0				
Normalized Distance	0.66				
Depth of Inheritance Tree (avg/max per type)		1.167	0.373	2	/Assignment 7/src/main/java/memoranda/EventsSc
Weighted methods per Class (avg/max per type)	490	16.333	17.904	71	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Children (avg/max per type)	0	0	0	0	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Overridden Methods (avg/max per type)	3	0.1	0.396	2	/Assignment 7/src/main/java/memoranda/TaskImpl
Lack of Cohesion of Methods (avg/max per type)		0.127	0.237	0.679	/Assignment 7/src/main/java/memoranda/TaskListIm
Number of Attributes (avg/max per type)	30	1	1.095	4	/Assignment 7/src/main/java/memoranda/TaskListIm
Number of Static Attributes (avg/max per type)	29	0.967	2.008	7	/Assignment 7/src/main/java/memoranda/History.ja
Number of Methods (avg/max per type)	181	6.033	7.889	37	/Assignment 7/src/main/java/memoranda/TaskImpl
Number of Static Methods (avg/max per type)	60	2	4.227	17	/Assignment 7/src/main/java/memoranda/EventsMa
Specialization Index (avg/max per type)		0.068	0.359	2	/Assignment 7/src/main/java/memoranda/Start.java
Number of Classes	30				
Number of Interfaces	0				
Total Lines of Code	2060				
Method Lines of Code (avg/max per method)	1257	5.216	5.445	33	/Assignment 7/src/main/java/memoranda/EventsMa

Figure 3 after Task 3

2.4. Static Methods decreased by 1, and the total LOC was reduced by 8. This was without fully refactoring, but commenting out the empty eventsChanged() method calls/definitions throughout the program.