AUTHORS
Matthew Garmon
Walter Conway

PURPOSE
This document outlines the instructions for running and testing the program with JUnit with the Eclipse IDE.

RUNNING THE PROGRAM with the JAR
The program was written using the Java programming language with the ECLIPSE IDE. Thus you will need the JRE Java Runtime Environment. The programming was packaged in a .JAR file, so you only need to double click it to run. Once this happens Figure 1 should appear.

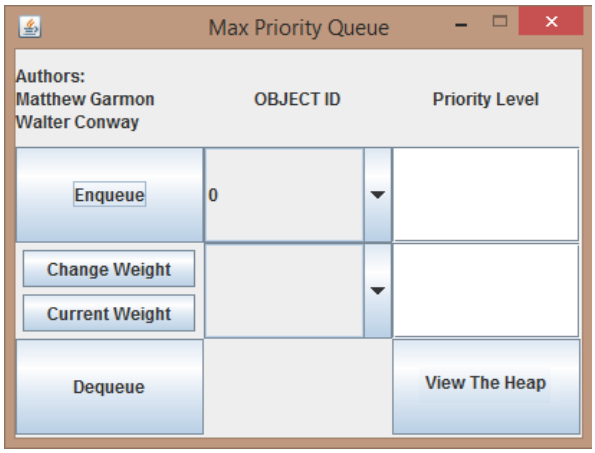


Figure 1

PARTS OF THE PROGRAM

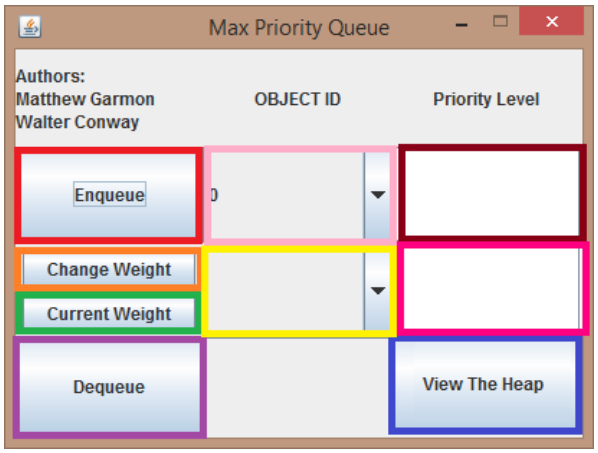


Figure 2

The colored areas represent the different points of interactions that the user is able to do with the interface.
The two labels “OBJECT ID” and “Priority Level” that are above the combo boxes and the two text fields represents what information that resides in those fields.

HOW TO ENQUEUE
To en-queue an object id you will need to select a valid object id from the combo box, which is outline in the lite pink above in figure 2. Once the action above is done you will see figure 3.

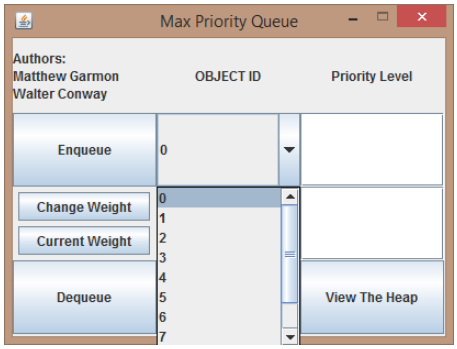
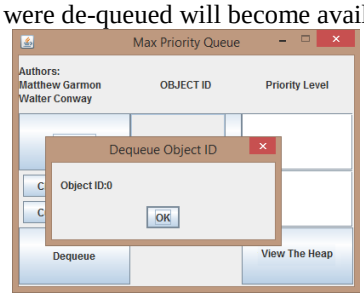


Figure 3

You will proceed to enter a priority level for the selected object id. And press the "Enqueue" button, which is outline in dark red in figure 2. Once this happens the object id from the combo box will disappear and become an option to either de-queue or change or check the current weight of an object id.

HOW TO DEQUEUE

To de-queue an object id you first must have an object id en-queued. If you have an object id en-queued by pressing the "Dequeue" button, which is outlined in purple in figure 2, a dialog box will appear notifying you of which object id was de-queued. As well as the ids that were de-queued will become available selections for en-queuing.



HOW TO CHANGE WEIGHT

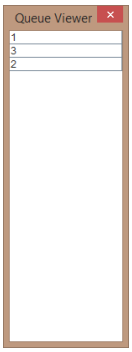
To change the weight of an object id you first must have an object id en-queued. If you have an object id en-queued by selecting the object id from the combo box outlined in yellow in figure 2, you can then type an appropriate priority level in the text field outlined in hot pink in figure 2. Then lastly, clicking the "Change Weight" button outlined in orange in figure 2, this will change the current weight that was assigned to that object id to whatever the valid priority level you changed it to.

HOW TO CHECK WHAT THE CURRENT WEIGHT IS

Checking the current weight of an object id is a similar process of changing the weight the two differences is that you do not have to type in the text field and you don't click change weight. You must select the object id from the combo box and click the "Current Weight" button, which is outlined in green in figure 2. Once this is done the current weight will appear in the text field of where you set the valid priority level in changing weight section. This text field is outlined in hot pink.

VIEW THE HEAP STRUCTURE

To view the heap structure that is the underlying data structure of the priority queue click the "View The Heap" button that is outlined in figure 2 by blue color outline. This is only accessible if there are items that have been en-queued. Once this is done a dialog box containing the heap structure will appear, as shown to the right.

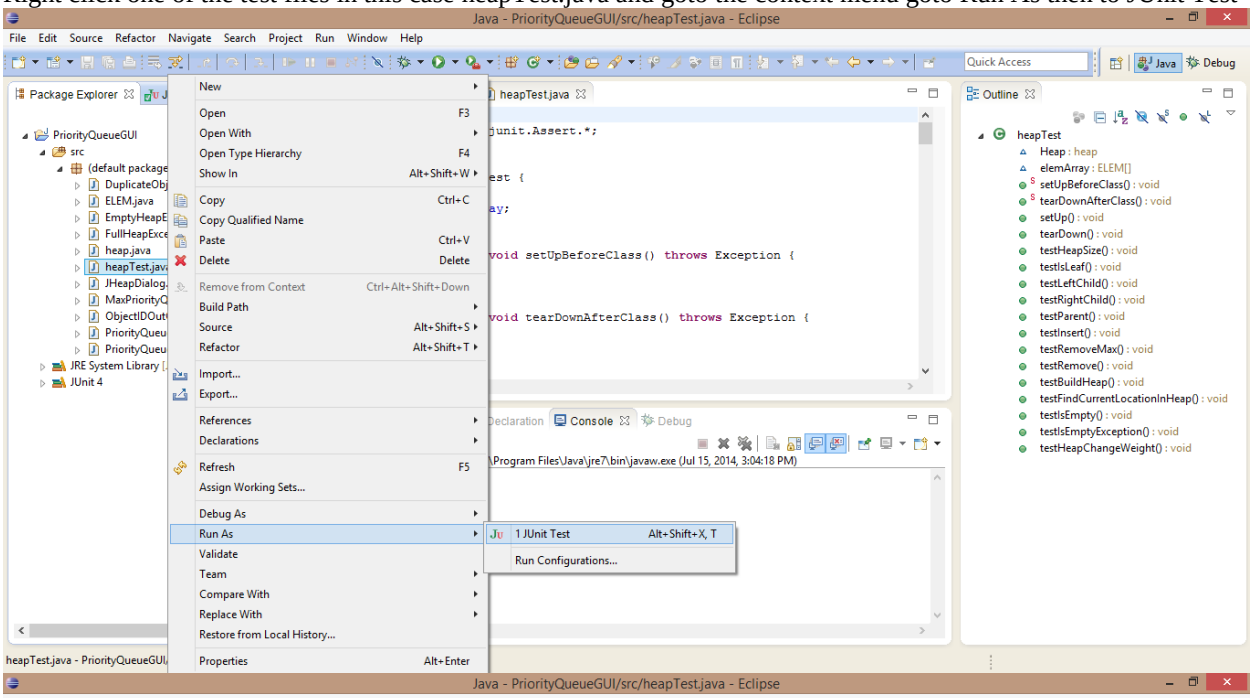


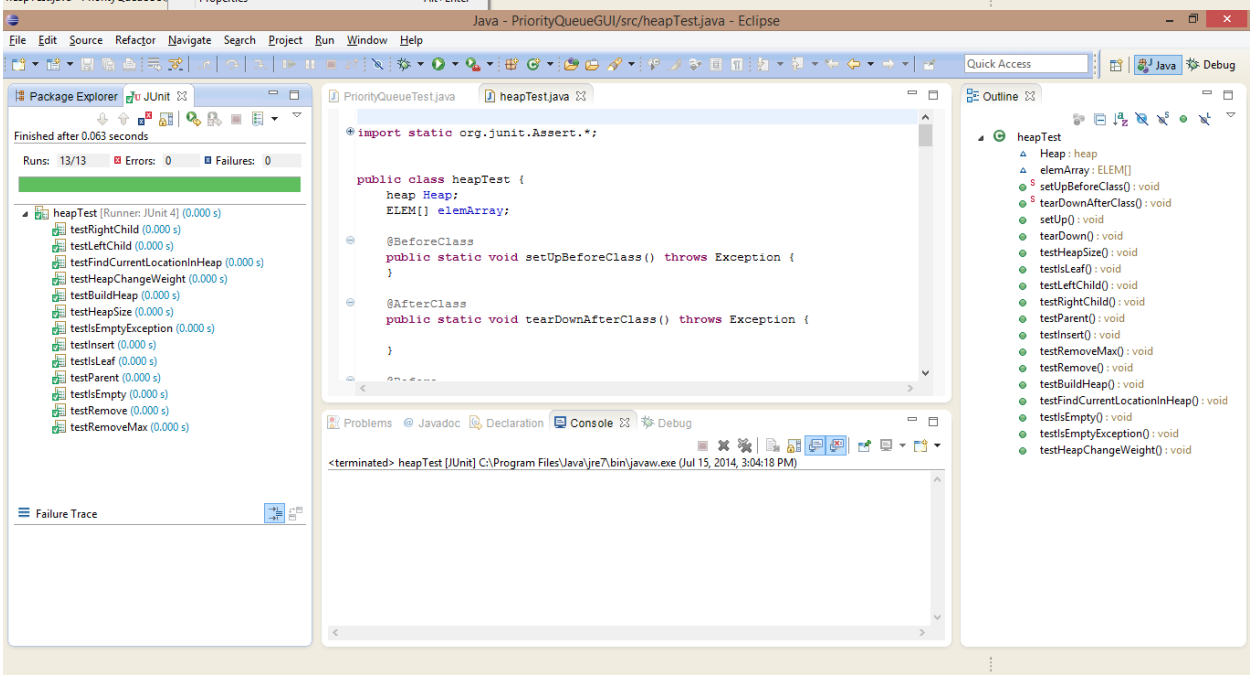
TESTING THE PROGRAM AND RUNNING THE CODE IN ECLIPSE

We have used JUnit version 4.0 for Testing. The files associated with this are PriorityQueueTest.java and heapTest.java, these files are included in the source package. The above two files and the source are packed in a file called workspace.zip. What you will need to do is unpack this then when you run eclipse it will ask you for a workspace this is where you will designate the unpacked path.

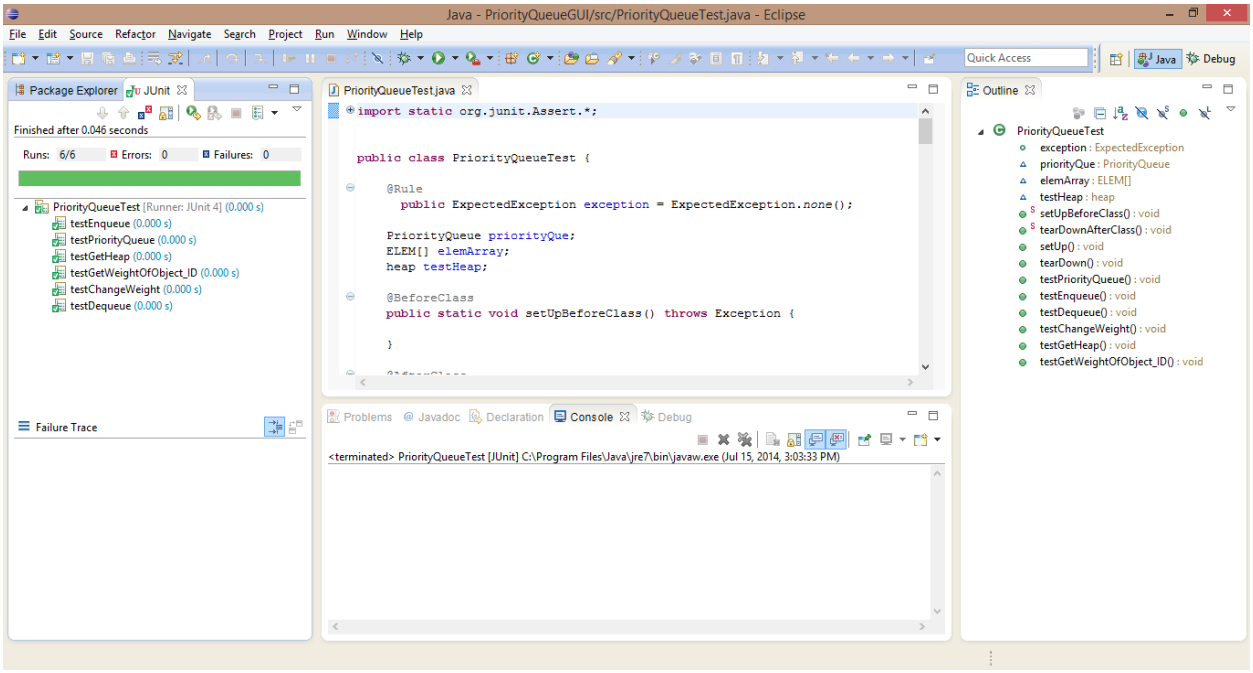
SCREEN SHOTS OF RUNNING TEST CASES

Right click one of the test files in this case heapTest.java and goto the context menu goto Run As then to JUnit Test





Like wise with the next File


Java - PriorityQueueGUI/src/PriorityQueueTest.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Java
Debug
Package Explorer
JUnit
Finished after 0.046 seconds
Runs: 6/6
Errors: 0
Failures: 0
PriorityQueueTest [Runner: JUnit 4] (0.000 s)
testEnqueue (0.000 s)
testPriorityQueue (0.000 s)
testGetHeap (0.000 s)
testGetWeightOfObject_ID (0.000 s)
testChangeWeight (0.000 s)
testDequeue (0.000 s)
Failure Trace
PriorityQueueTest.java
import static org.junit.Assert.*;
public class PriorityQueueTest {
@Rule
public ExpectedException exception = ExpectedException.none();
PriorityQueue priorityQue;
ELEM[] elemArray;
heap testHeap;
@BeforeClass
public static void setUpBeforeClass() throws Exception {
}
Problems
Javadoc
Declaration
Console
Debug
<terminated> PriorityQueueTest [JUnit] C:\Program Files\Java\jre7\bin\javaw.exe (Jul 15, 2014, 3:03:33 PM)
Outline
PriorityQueueTest
exception : ExpectedException
priorityQue : PriorityQueue
elemArray : ELEM[]
testHeap : heap
setUpBeforeClass() : void
tearDownAfterClass() : void
setUp() : void
tearDown() : void
testPriorityQueue() : void
testEnqueue() : void
testDequeue() : void
testChangeWeight() : void
testGetHeap() : void
testGetWeightOfObject_ID() : void