COMP3111: Software Engineering

Debugging Tools

Learning Outcomes

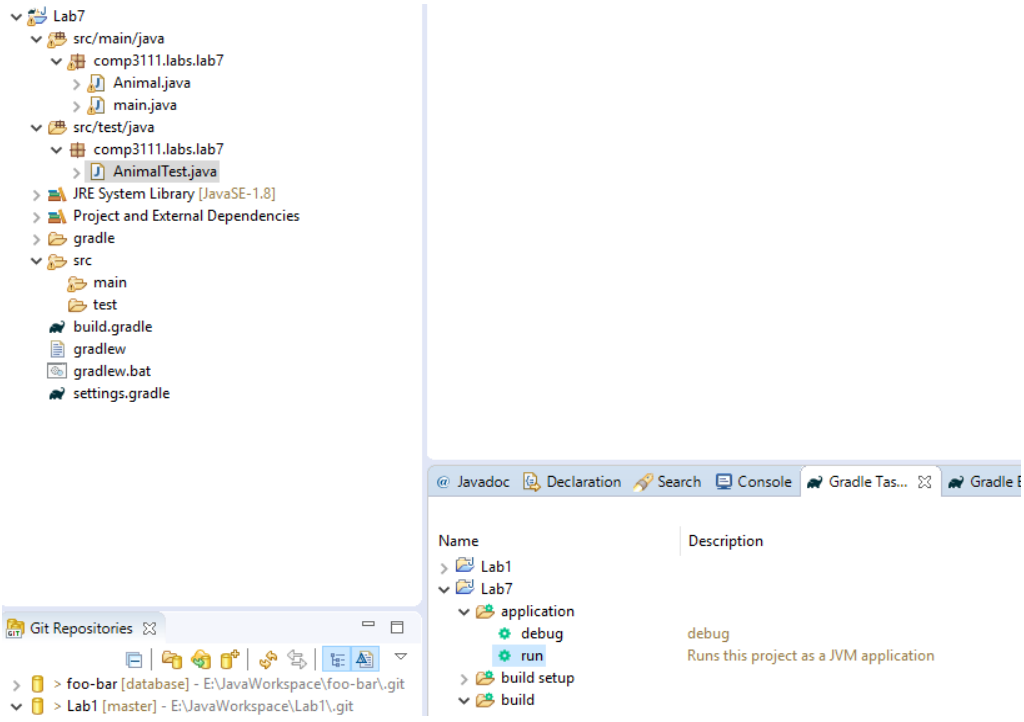
- Be able to debug using the Eclipse debugger

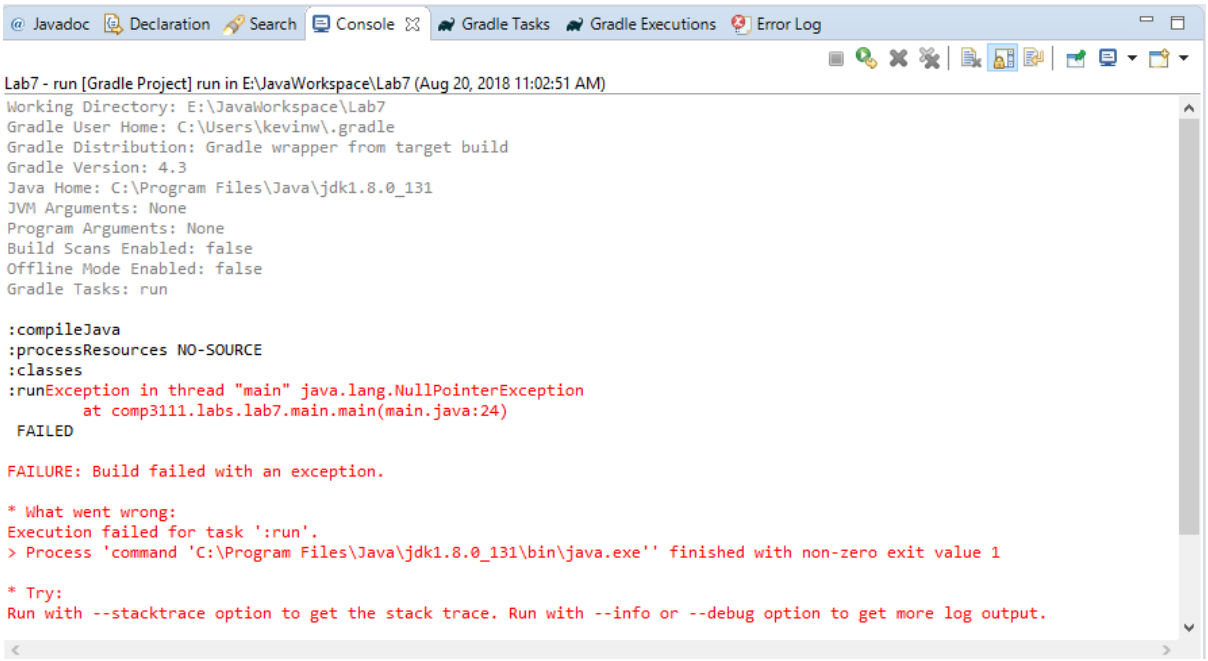
Setup: Clone the Unit Testing lab repository from https://github.com/khwang0/comp3111-lab7 into Eclipse.

(Despite it is called lab7, this is the lab we want you to do this week 😉)
Exercise 1: Locate and fix a bug

Step 1.1: Go to the Gradle Task windows and click run. You should encounter some errors.

Note: if you cannot find the Gradle task windows, open it from the the menu bar > Windows > Show View > Others.

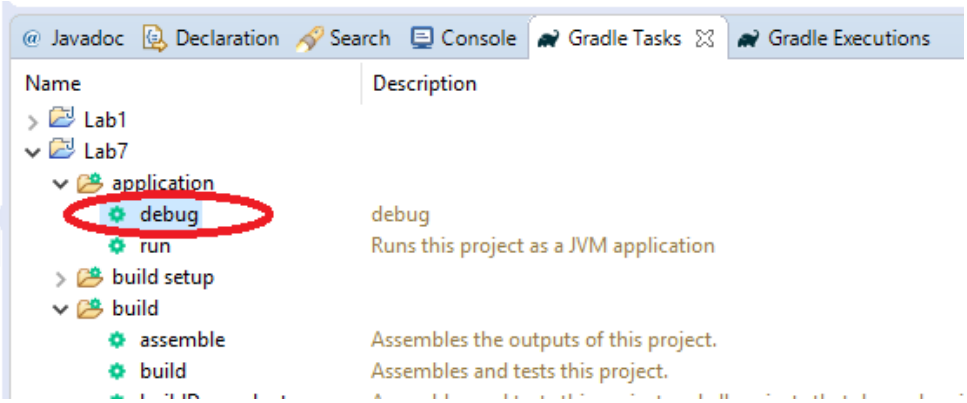




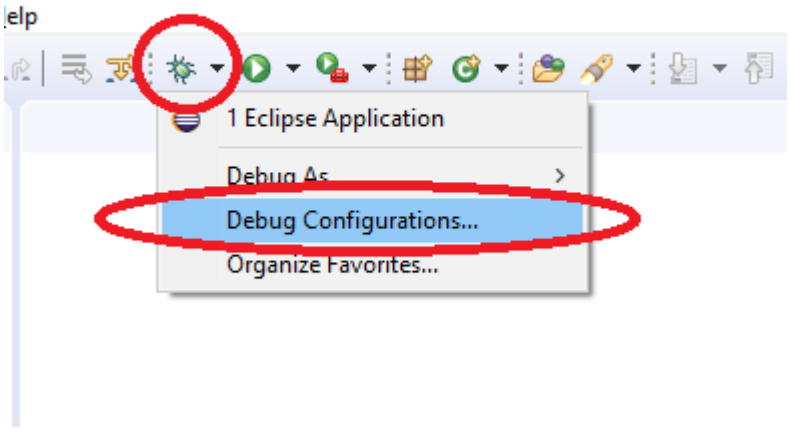
Despite this time the console states pretty clear what has happened, sometime you can have a very lengthy error in Java. So we try to enable the debugger by the following ways

Step 1.2: Go to the Gradle Task windows and click debug. The program is now prepared to be debugged.

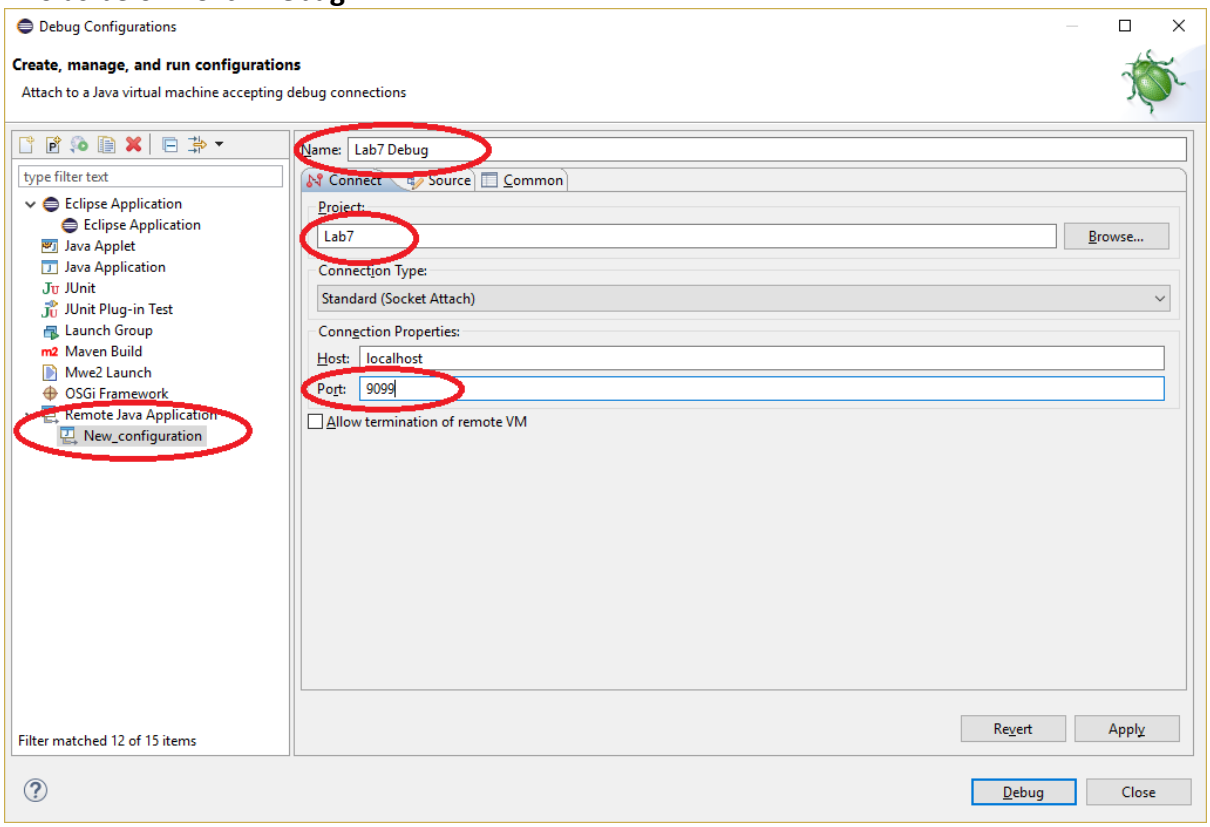
Note: this debug task is a user created task and it does not come with the default setting. In this task we bind the port 9099 for a debugger to attach to. For details, please look at the build.gradle.



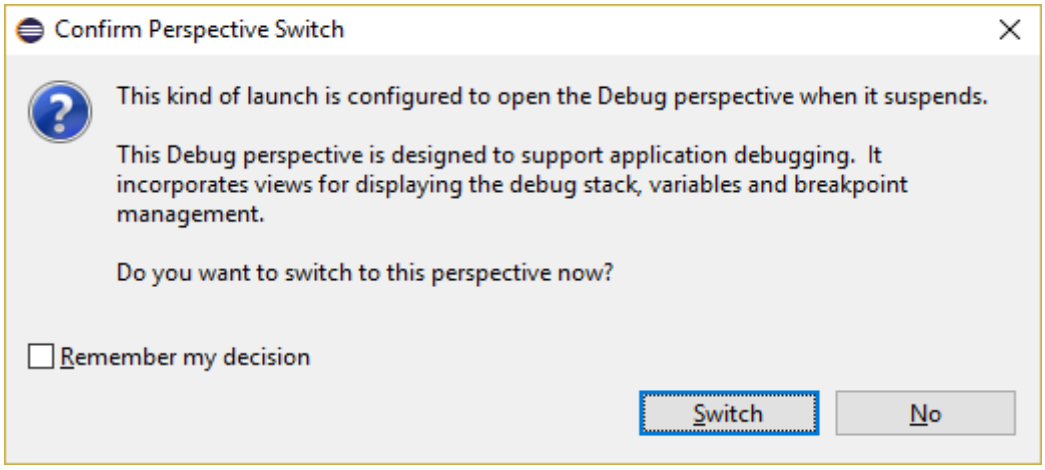
Step 1.3: Click the Debug icon on the menu bar and select Debug Configurations.



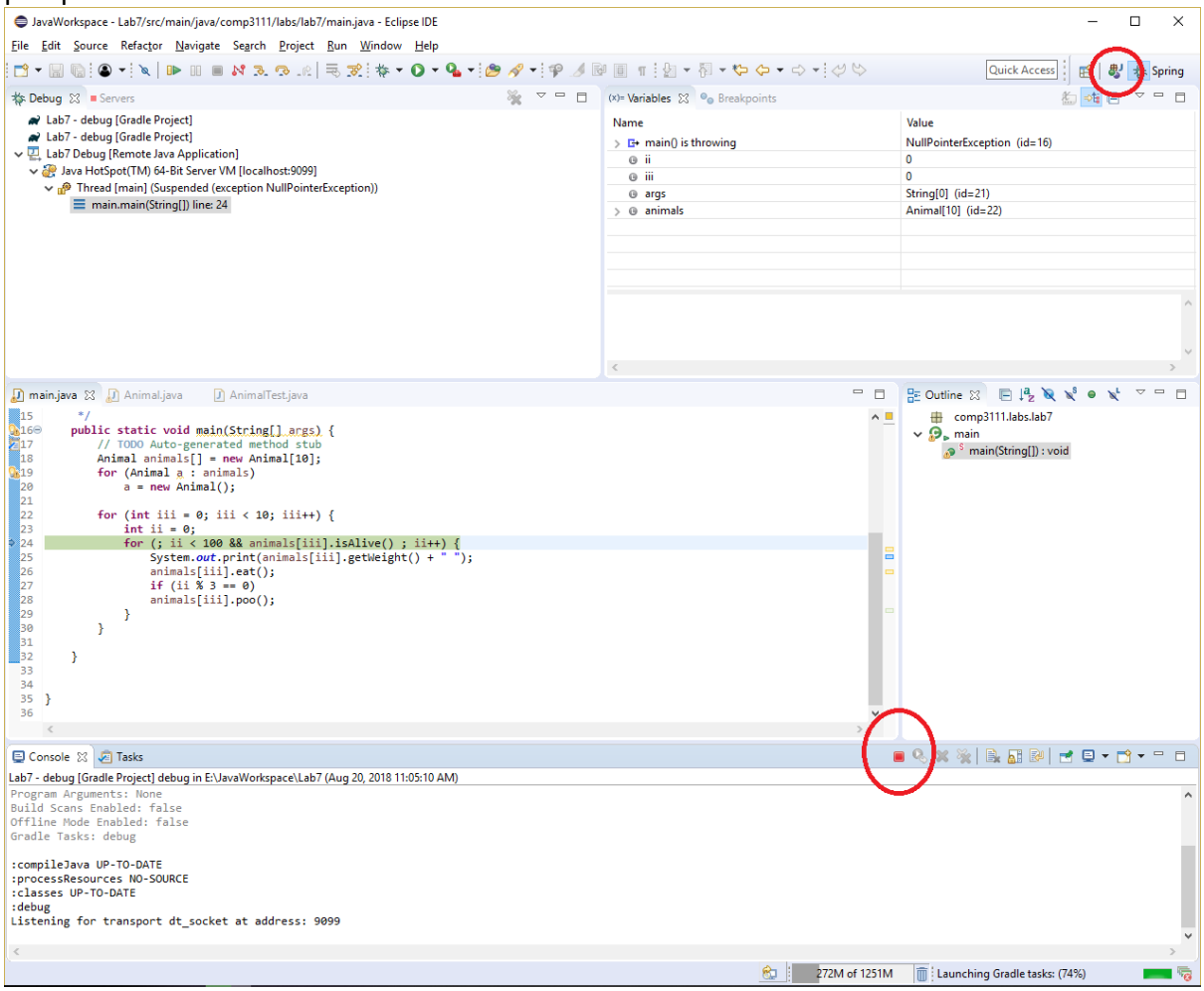
Step 1.4: Double click on “Remote Java Application” to add a new configuration. Type the info as below. Click **Debug**.



Step 1.5: You will see the following dialog, click “Switch”. This will change to the debug perspective.



Now you should see an entire different screen and pause the program at the point of the error. **Click stop button** and **click the icon that circled below** to switch back to the Java perspective.



Step 1.6: Now we need to insert a breakpoint to your program. We know the program will be stopped at line 25 or main.java. To assure the constructor of Animal has been executed, we insert a breakpoint inside the constructor of Animal.java.

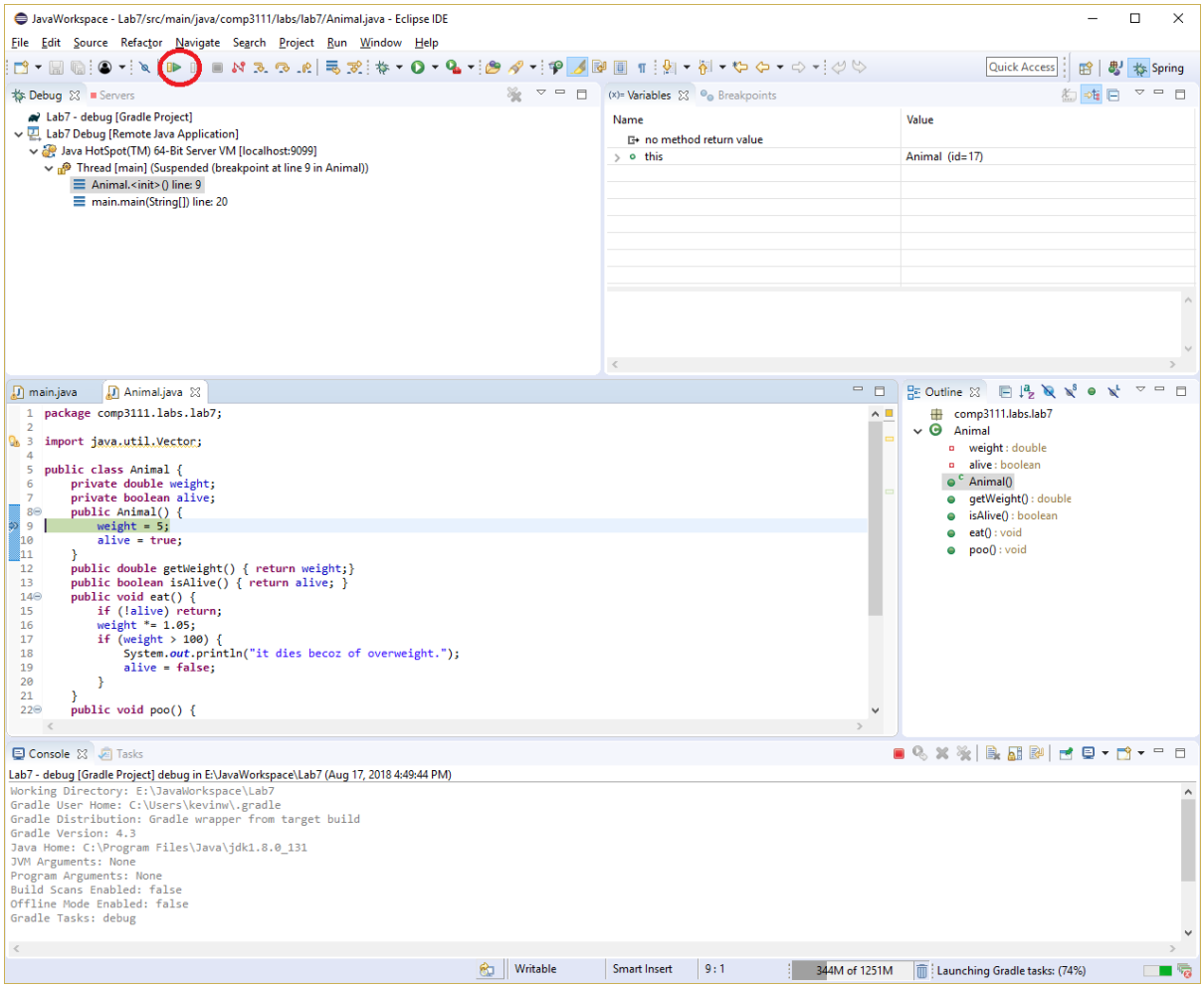
Open Animal.java and double click the line-number 9 to insert a breakpoint there.

main.java | Animal.java

```
1  package comp3111.labs.lab7;
2
3  import java.util.Vector;
4
5  public class Animal {
6      private double weight;
7      private boolean alive;
8      public Animal() {
9          weight = 5;
10         alive = true;
11     }
12     public double getWeight() { return weight;}
13     public boolean isAlive() { return alive; }
14     public void eat() {
15         if (!alive) return;
16         weight *= 1.05;
17         if (weight > 100) {
18             System.out.println("it dies becoz of overweight.");
19             alive = false;
20         }
21     }
22     public void poo() {
23         if (!alive) return;
24         weight -= 1;
25         if (weight < 0) {
26             System.out.println("it dies becoz of underweight.");
```

Note: Your program will stop at a breakpoint when you run it in debug mode. To remove the breakpoint, simply double click it again.

Step 1.7: Redo Step 1.2 and rerun the Debug from menu bar, your program should be stopping at the breakpoint. You can click the resume button (F8) to continue the program. After clicking for 10 times, you will encounter the same error. That means the bug is still there.


JavaWorkspace - Lab7/src/main/java/comp3111/labs/lab7/Animal.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Spring
Debug
Servers
Lab7 - debug [Gradle Project]
Lab7 Debug [Remote Java Application]
Java HotSpot(TM) 64-Bit Server VM [localhost:9099]
Thread [main] (Suspended (breakpoint at line 9 in Animal))
Animal.<init>() line: 9
main.main(String[]) line: 20
Variables
Breakpoints
Name
Value
no method return value
this
Animal (id=17)
main.java
Animal.java
1 package comp3111.labs.lab7;
2
3 import java.util.Vector;
4
5 public class Animal {
6 private double weight;
7 private boolean alive;
8 public Animal() {
9 weight = 5;
10 alive = true;
11 }
12 public double getWeight() { return weight;}
13 public boolean isAlive() { return alive; }
14 public void eat() {
15 if (!alive) return;
16 weight *= 1.05;
17 if (weight > 100) {
18 System.out.println("it dies becoz of overweight.");
19 alive = false;
20 }
21 }
22 public void poo() {
Outline
comp3111.labs.lab7
Animal
weight : double
alive : boolean
Animal()
getWeight() : double
isAlive() : boolean
eat() : void
poo() : void
Console
Tasks
Lab7 - debug [Gradle Project] debug in E:\JavaWorkspace\Lab7 (Aug 17, 2018 4:49:44 PM)
Working Directory: E:\JavaWorkspace\Lab7
Gradle User Home: C:\Users\kevinw\.gradle
Gradle Distribution: Gradle wrapper from target build
Gradle Version: 4.3
Java Home: C:\Program Files\Java\jdk1.8.0_131
JVM Arguments: None
Program Arguments: None
Build Scans Enabled: false
Offline Mode Enabled: false
Gradle Tasks: debug
Writable
Smart Insert
9 : 1
344M of 1251M
Launching Gradle tasks: (74%)

Lab Activity and Assessment

Lab Activity

1) Able to use the debugger and insert a breakpoint anywhere you want.
2) Fix the bug in the main.java so that the program ends outputs

```
:compileJava
:processResources NO-SOURCE
:classes
:run
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628
5.0 4.25 4.4625 4.685625000000001 3.9199062500000013 4.115901562500001 4.321696640625001 3.5377814726562518 3.71467054628

BUILD SUCCESSFUL in 0s
2 actionable tasks: 2 executed
```

,where each line ends with

… `0.5353828341528297 0.5621519758604713 it dies becoz of underweight.`

Assessment

Demo this at lab or submit the following on Canvas. Missing any part may yield a 0.

1. The fixed code of Main.java. It should have the comment:
    a. The line number of where the bug was.
    b. Correctly explain why there was a bug there. We are expecting your explanation including terms like pointer/reference/object.