MIT Open Learning Library

My Courses   All Courses   About   Contact   ♥ Give   Help   **vitorpbarbosa7**

Course   Progress

Course › Readings/Videos › Reading 3: Testing › Questions

‹ Previous   [icons]   Next ›

Questions

🔖 **Bookmark this page**

---

coverage

3/3 points (graded)

Install EclEmma in Eclipse.
Then create a new Java class called Hailstone.java containing this code:

```java
public class Hailstone {
  public static void main(String[] args) {
    int n = 3;
    while (n != 1) {
        if (n % 2 == 0) {
            n = n / 2;
        } else {
            n = 3 * n + 1;
        }
    }
  }
}
```

Run this class with EclEmma code coverage highlighting turned on, by choosing Run / Coverage As / Java Application.

By changing the initial value of n, you can observe how EclEmma highlights different lines of code differently.

When n=3 initially, what color is the line n = n/2 after execution?

[green ▾]   ✔ **Answer:** green

Explanation
Run EclEmma and find out!
If it's hard to perceive the difference between the red and green highlighting, you can change the color of the coverage highlighting by going to Preferences / General / Appearance / Editors / Text Editors / Annotations. The annotations to change are called Full Coverage, Partial Coverage, and No Coverage. You might, for example, make Full Coverage white, Partial Coverage light gray, and No Coverage dark gray.

When n=16 initially, what color is the line n = 3 * n + 1 after execution?

[red ▾]   ✔ **Answer:** red

Explanation
Run EclEmma and find out!

What initial value of n would make the line while (n != 1) yellow after execution?

[1]   ✔ **Answer:** 1

[1]

Explanation
Yellow is used for a line that contains a branch (e.g., an if or while predicate) that is only taken in one direction during the program -- either the predicate is false every time the program gets there, or the predicate is true every time. In this case, the way to make the while predicate yellow is to make n != 1 false, so that the predicate executes just once and the loop is skipped entirely. n=1 does that.

[Submit]   ⓘ Show Answer

---

❶ Answers are displayed within the problem

‹ Previous   Next ›

Open Learning Library

About

Accessibility

All Courses

Why Support MIT Open Learning?

Help

Connect

Contact

Twitter

Facebook

Privacy Policy   Terms of Service