Debugging Instrumented Code

We are working on adding more tooling for debugging your instrumented code. At the moment there are several guidelines to help you if you get stuck:

1. Check out theUnderstanding Instrumented Code
2. page to understand how Scribble instruments.
3. Use descriptive user-readable labels and look at emitted events to understand which property failed.
4. If you are using event-based assertions mode, you can use the--debug-events
5. option, which will emit an additional event for each failure that holds the values of primitive variables appearing in the event.
6. If you have added contract-wide invariants and your contracts are failing to be deployed, be aware that contract-wide invariants are checked at the end of constructors. So if your invariant doesn't hold at the end of the constructor, it will prevent it from being deployed. If you are having trouble deploying instrumented contracts with invariants, you should also read the section onexternal calls in invariants
7. .
8. Be aware of cases where your invariant evaluation itself may cause a failure. Specifically if you are using the division opreator or calling somepure
9. /view
10. function which may fail under some conditions.
11. If scribble is generating code that doesn't compile or generating otherwise invalid code, feel free to reach out andfile an issue
12. .
13. You can also find us on the #scribble channel of theMyhtX Discord Server
14. .
15.

Previous Instrumented Code Next JSON output specification Last updated3 years ago

On this page Was this helpful?