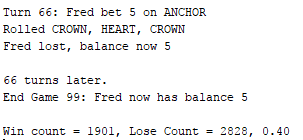
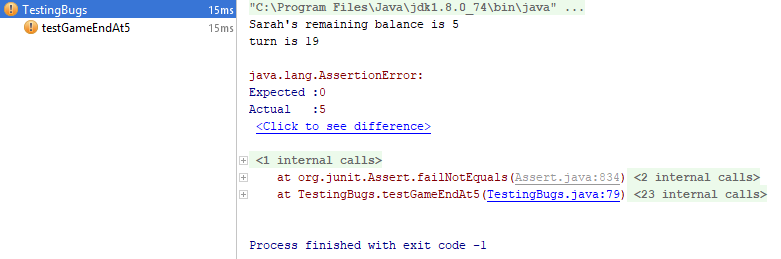
Debug log for bug “Betting Limit Bug”

Initial example of bug



Output from automated test



Log:

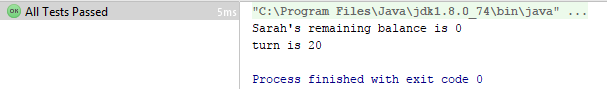
|  |  |
| --- | --- |
| **Step reasoning** | The origin of the issue is that the game ends without hitting the limit. Let’s start by verifying the UI is printing the correct value. |
| **Hypothesis 1** | The UI is printing the incorrect value of the players remaining balance after the game ends |
| **Test** | Debug mode and view the value of ‘player.getBalance()’ |
| **Prediction** | Value returned by Player.getBalance is different from the one shown in UI |
| **Result** | Prediction incorrect, print statement is correct, value is infected |

|  |  |
| --- | --- |
| **Step reasoning** | he issue is the game is quitting prematurely, what decides when the game ends?  There is a while loop which re-runs the game while the player is above his limit, and below twice their initial balance. There are two conditions, once checks if we’ve going to exceed the limit, and one that check if we’ve made more than the target. It’s going to be an issue with reaching the limit. |
| **Hypothesis 2** | The value passed into player.balanceExceedsLimitBy(bet) in Main is infected |
| **Test** | Debug mode and view the value of ‘bet’ |
| **Prediction** | Bet is incorrect |
| **Result** | Bet’s value is declared only once in the code, then never changed. Bet’s value is always sane. |

|  |  |
| --- | --- |
| **Step reasoning** | Step into balanceExceedsLimitBy(bet) |
| **Hypothesis 3** | Method balanceExceedsLimitBy(bet) is not meeting its method contract. |
| **Test** | Debug mode, go into the method, verify all values going in are correct, and value coming out is correct |
| **Prediction** | Method contract is being violated |
| **Result** | Input value is correct; output value is infected. |

|  |  |
| --- | --- |
| **Step reasoning** | There is only one line in this method! It must be the issue. Thinking about it, in the line ‘(balance - amount > limit)’, if I used (5-5 > 0), it would return false, as 0 is not greater than 0. |
| **Hypothesis 4** | This is a relational operator issue |
| **Test** | Change the greater than symbol to a greater or equal to symbol, and rerun the automated test, it will pass |
| **Prediction** | Issue will be resolved |
| **Result** | Issue is resolved |

Successful automated test result



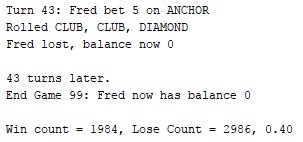
Initial bug, /src/Player.java, line 30



Resolved bug, /src/Player.java, line 30



Resolved bug, UI output



Regression testing:

Re-running previous tests to ensure no new bugs have been created.

