| **Test Name** | | | Game does not pay out at the correct level | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Tested:** | | | Crown and Anchor Game | | | |
| **Test Description:** | | | If the player wins on 1 match, the players balance will not increase | | | |
| **Pre-conditions** | | | Default game settings | | | |
| **Post-conditions** | | | After each match the player is able to check the output to determine if the balance is correct | | | |
| **Notes:** | | **At present a pass indicates that the bug does exist within the game. This is the expected behavior in order to confirm the bug is really a bug** | | | | |
| **Result (Pass/Fail/Warning/Incomplete)** | | **Fail – Indicates the bug does exist** | | | | |
|  | **TEST STEP** | | | **EXPECTED TEST RESULTS** | P | F |
|  | Run program | | |  | x |  |
|  | Search through the output from the game and locate the first instance of the player winning a match | | | Confirm the balance that the player had before the match started does not increase | x |  |
|  |  | | |  |  |  |

# Test Result

Output from game

Rolled CLUB, ANCHOR, DIAMOND

Fred lost, balance now 95

Turn 2: Fred bet 5 on CROWN

Rolled CLUB, ANCHOR, DIAMOND

Fred lost, balance now 90

Turn 3: Fred bet 5 on CLUB

Rolled CLUB, ANCHOR, DIAMOND

Fred won 5, balance now 90

Diagnose

From the above output from the game we can see that after three match’s the player had a win. Once the player won a match we can see that his balance did not increase as expected. The output indicates that the player won 5, which should have given his balance a total of 95; he had 90 due to the first to lose of the game. This test confirms that the bug does actually exist within the game.

From looking through the output of the game the balance calculation seems to be the issue, the game is subtracting correctly when the player loses but is failing to add to the balance on a win. Instead it is just keeping the balance as it was

The output below confirms my diagnoses as the player won a few games in a row and never increased his balance.

Extra output to confirm diagnoses

Turn 4: Fred bet 5 on HEART

Rolled CLUB, ANCHOR, DIAMOND

Fred lost, balance now 85

Turn 5: Fred bet 5 on ANCHOR

Rolled CLUB, ANCHOR, DIAMOND

Fred won 5, balance now 85

Turn 6: Fred bet 5 on CLUB

Rolled CLUB, ANCHOR, DIAMOND

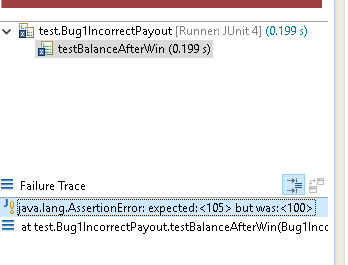
Fred won 5, balance now 85

Turn 7: Fred bet 5 on CLUB

Rolled CLUB, ANCHOR, DIAMOND

Fred won 5, balance now 85

# Automated Test with Junit to show the bug



The screenshot above shows the results of a Junit test that automates the bug, the fail of the Junit test indicates that the bug does exist and the program is not increasing the players balance correctly on a win. In order to indicate the bug within an automated process the Junit test fails on checking the players balance. Even though the fail occurs on the check balance method this does not indicate where the actual bug is. As the bug would be happening more likely to happen when the program calculates the value for a win/lose