Comment: Although SolidiFI claims to have injected 1737 integer overflow anomalies into 50 contracts, the 1737 anomalies are only replicas of 40 pre-made anomaly fragments, which are as shown in the following figures(The following code snippets are all the same, but the names of variables are different).





















Comment: There are anomaly injection fragments with issues that prevent the injected anomalies from being activated.



Fig.1 : An example of incapability of SolidiFI to accurately inject and precisely label anomalies. SolidiFI does not insert any statement into the contract to modify the value of the variable *balances\_intou38* (declared in line 1). This makes *balances\_re\_ent38[msg.sender]* keep the initial value (0) unchanged. If the *require-statement* (line 3) is to be true, \_*value* must be equal to 0 at this point. If *\_value* > 0, it will result in dead code in the statements on line 4 and line 5. The above 10 fragments all have the above issues.