**Software Testing Algorithm**

1. Management team G generates (Ci,k)a, from Ti to Tk,

1 ≤ k ≤ N for an Sa, 1 ≤ a ≤L. Repeat this process for Sa , a = 1, 2, ...L.

1. Tk produces M/4 results where M is the total number of results to be produced by the smart contract for software testing as per assignment in the (Ci,k)a.
2. If the result Rt ,1 ≤ t ≤ M/4, generated by Tk meets the

acceptance criteria of (Ci,k)a, Rt is added to the blockchain as

a blockchain transaction.

1. If Rt generated by Tk does not meet the acceptance

criteria of (Ci,k)a, Ti is alerted by the blockchain that Rt is not

successful, and the fault percentage of the result is calculated and stored in the transaction node.

Here Accuracy percentage is calculated as:

**AP = (Output Received) / (Ideal Output Mentioned In C(i,k) ) x 100**

* **Value of X is initialized as 0.**

1. The value of P is extracted from the smart contract and the highest Accuracy percentage out of the lowest P% of the results that don’t meet the acceptance criteria of the C(i,k)a will be stored as the value X. and the remaining results will be added to blockchain with the rejection message.
2. All results of the smart contract will be generated and the results whose accuracy percentage falls below the X value will be dumped.
3. Repeat steps 2 to 5 for, Tk, 1 ≤ k ≤ N.

Test Cases:

M=1000, P=20%, AC = Acceptance Criteria, AP = Accuracy Percentage

| Test Case ID | Description | Flow | Expected Result |
| --- | --- | --- | --- |
| TC1 | Out of the 250 results that’ll run, suppose 150 of the results fail to meet the acceptance criteria. | 1->2->3->4->5->6->7…  When the above flow is attained it is determined that the value of X supposedly turns out to be 16%. | Out of all the M results, the ones that fail to meet the AC and having AP lower than 16% will be dumped and the rest will be added to the blockchain. |
| TC2 | Out of the 250 results that’ll run, suppose all of them pass the acceptance criteria. | 1->2->3….  When the above flow is attained, the value of X will be equal to its initialised value. | All the results will be directly added to the blockchain and none of them will pass validation cycle. |
| TC3 | Out of the 250 results that’ll run, suppose all the 250 of the results fail to meet the acceptance criteria. | 1->2->4->5->6->7…  When the above flow is attained it is determined that the value of X supposedly turns out to be 18%. | Out of all the M results, the ones that fail to meet the AC and having AP lower than 18% will be dumped and the rest will be added to the blockchain. |
| TC4 | Out of the 250 results that’ll run, suppose 1 of the results fail to meet the acceptance criteria. | 1->2->3->4->5->6->7…  When the above flow is attained it is determined that the value of X supposedly turns out to be 27%.(Here the value of X will automatically be the AP of the 1 failed result) | Out of all the M results, the ones that fail to meet the AC and having AP lower than 27% will be dumped and the rest will be added to the blockchain. |