**True Positive and False Positive Calculation’s Method:**

CWE476 Null Pointer Dereference test cases of Juliet Test Suite consist of flawed and unflawed test cases, named as *Positives* and *Negatives* in our related research.

The test cases have file groups based on *name, numbering*, and *suffix*. Juliet sometimes uses the concept of a data source and a data sink to divide the chain of events leading to a flaw. The naming convention of Juliet’s *Positives* uses bad() for the entry into a flaw, and this may, in turn, use a badSource() and badSink() pattern. It also uses good() for representing the *Negatives*, which in turn consist of goodSource() and goodSink().

Detecting a test case within a line inside a bad(), badSink() or badSource() considered a true positive (TP). Consequently, detections within a line inside a good(), goodSink() or goodSource() considered a false positive (TP)