Some pairs of test cases are created for the purpose of checking different cases.

1. **test_1 and test_2 :**

   Only return values for the else condition are changed. So, they are not equivalent. But only one constraint is different in them, others are equivalent. The equivalent pairs would be in the matched file, the non-equivalent pairs would be in the unmatched file. And the non-matched file contains the constraints from both the programs which are not equivalent.
2. **test_3 and test_4 :**

   They are almost identical like test_1 and test_2 but the “else if” condition is changed. Similarly like the last testcases, they are not equivalent and the results are shown in the three files.
3. **add and add_processed :**

   add_processed is the optimized version of the add program. That means redundant blocks are removed from the original add program. The add.c file is present in the testSource folder. The flow function contains different branch conditions but it is called from the main function with constant values of the parameters. So, it is only possible to execute only one branch. In the optimized version, the other branches can be removed. This optimization will reduce the 15 path constraints to 3.

   While collecting the path constraints from the binary files, we only target the flow function leaving the others. In the matched file, the matched constraint will be saved and rest in the unmatched file & non-matched file.