Chapter 8

Comparison of POSAR and AFC test results

8.1 Steps to manually compare the POSAR and AFC test results

The effectiveness of Planning-Operationalisation-Separation-Analysis-Reporting (POSAR) test over Abstraction-Filtration-Comparison (AFC) test is the matter of concern of this chapter. This cannot be done mathematically, probabilistically or by using software tools in an objective manner because both AFC and POSAR are subjective procedures and so, do require a subjective comparison procedure. One of the subjective ways (possibly more intelligible to the judiciary) of checking the effectiveness of POSAR over AFC is to carry out the following steps in sequence.

Step 1: Take software A (which can be the original software\textsuperscript{14} in the case.)

Step 2: Take software B (which can be the pirated software\textsuperscript{15} or the allegedly pirated software)

Step 3: Compare A and B using POSAR and present the results (C) in the prescribed format.

Step 4: Compare A and B using AFC and present the results (D) in the prescribed format.

Step 5: Compare C and D.

Step 6: If C contains more pieces of evidence than D then POSAR is more effective than AFC

Step 7: If C contains the same pieces of evidence as D then POSAR and AFC are equally effective.

Step 8: If C contains fewer pieces of evidence than D then POSAR is less effective than AFC

\textsuperscript{14} Throughout this thesis, original means the version of the software that the complainant submits to the law enforcement agency for software piracy forensics. This thesis presupposes that the law enforcement agency has satisfactorily verified the legal aspects of the documentary evidence of copyright produced by the complainant and is convinced that the complainant is the copyright holder of this version of the alleged software.

\textsuperscript{15} Throughout this thesis, pirated means the allegedly pirated software
8.2 Result reporting features of AFC and POSAR juxtaposed

AFC test does not provide a structure to present the final results (at the end of its comparison phase). Unlike AFC, POSAR has a reporting phase which helps the cyber forensic expert to present the results in structured manner. The results of POSAR are available in the form of two lists and six sets. They are; (1) One list of similarities / commonalities between the original and the pirated coupled with another list of piracy of thumb impressions, programming errors, programming blunders etc. (2) a second list of suspected post-piracy modifications along with the idealized, pre-modified form of these suspected post-piracy modifications; (3) SP1, a abstracted, localized set of the pirated; (4) SP2, a set of idealized, pre-modified form of fields of the suspected post-piracy modifications in the pirated; (5) SP3, a set of suspected post-piracy modifications in the pirated; (6) SO1, the abstracted, localized set of the original; (7) SO2, a set of idealized, pre-modified form of fields of the suspected post-piracy modifications in the original; and (8) SO3, a set of suspected post-piracy modifications in the original. In addition to these, POSAR provides a report format (see Table 8.1, above) for presenting the results of data base level comparison, if any. All these together present the POSAR results concerning similarities and commonalities in numerical and non-numerical terms. In short, when AFC results are presented in expert-specific manner, the POSAR results are presented in expert-independent manner.
18. **Similarity in the table names**: ___% commonality.

19. **Length of the ‘original’ table**: a

20. **Length of the ‘pirated’ table**: b

21. **Percentage of similarity in lengths**: (a/b)*100

22. **Field count of the ‘original’**: c

23. **Field count of the ‘pirated’**: d

24. **Percentage of similarity in field count**: (c/d)*100

25. **Perfect commonality in the names of fields**: ___ out of ___ names of fields in the ‘original’ are found in ‘pirated’ also. So, ___% commonality

26. **Perfect commonality in name and data type among fields**: ___ out of ___ fields have the common name and data types. So, ___% commonality in name and data type.

27. **Perfect commonality in name, data type and length among fields**: ___ out of ___ fields have same names, data types and length. So, ___% commonality

28. **Perfect commonality in name, data type, length and the default values set in the fields**: ___ out of ___ fields have same names, data types, length and default values. So, ___% commonality.

29. **Perfect commonality in sequence of the fields with same name**: ___ out of ___ fields with same name, do occur in the same sequence. So, ___% commonality.

30. **Perfect commonality in sequence of the fields with same name, data type, length and default values**: ___ out of ___ fields (with same name, data type, length and default values) do occur in the same sequence. So, ___% commonality.

31. **Count of comparable (mappable) fields including suspected-post-piracy modified / created fields**: ___ out of ___ fields in the ‘pirated’ can be perfectly or approximately mapped (in terms of names) to at least one field in the ‘original’. So, ___% comparable fields in the ‘pirated’.

32. **Count of non-mappable but suspected-post-piracy fields, including ending fields**: ___ out of ___ fields in the ‘pirated’ could not be properly mapped to any of the fields in the ‘original’ but they can be suspected to be post-piracy modifications. So, ___% incomparable but suspected fields in the ‘pirated’

33. **Count of non-mappable, non-suspected fields**: ___ out of ___ fields in the ‘pirated’ could not be properly mapped to any of the fields in the ‘original’ and do not provide any clue to be suspected as post piracy modification. So, ___% incomparable, non-suspected fields in the ‘pirated’

34. **Inference**: Piracy is confirmed / largely suspected / loosely suspected / not suspected.

Table -8.1: The proposed format for presenting the result of comparison of two database tables (post-piracy modifications are also considered)
8.3 **Availability and nature of test data**

The surest way to establish the effectiveness of POSAR over AFC is to apply both in real life software piracy litigations and then compare the results. Carleton (1995) had tested his “Identification-Filtration-Comparison” (as an alternative test to AFC) in the real life case Lotus Development Corp. v. Borland International\(^\text{16}\) (Hamilton and Sabety, 1997, p.274), in which the US judiciary had already used AFC to establish software copyright infringement. Following Carleton’s approach, POSAR needs to be applied in a real life case (situation) in which AFC has already been applied with results published. But obtaining the AFC test results (that means, the expert report) of these real life cases is not easy, for many reasons. Firstly, AFC is an exclusively American method used only in America. Secondly, the AFC results are inaccessible except through American courts. So, the best possible way in the context of this work is to take real life software piracy (or software copyright infringement) cases from Indian judiciary and apply both POSAR and then the AFC in these situations with the objective of checking the effectiveness of POSAR over AFC. Thus such an instance from the researcher’s own experience (as the cyber forensic expert in software infringement cases) has been utilized here. However, the results of this comparison are intended to be used only for academic purpose of this thesis, are not for influencing or re-analysing or questioning the court’s judgment and have no legal or judicial implications.

8.4 **The test cases**

Applying AFC and POSAR in the entire software package can be a task possibly beyond the scope of this study and hence to simplify matters, AFC and POSAR can best be applied to certain areas (for instance, databases, program modules etc.) of the alleged software. As long as the areas are selected intelligently, carefully and expertly, as this study has tried to do, the results can be just as reliable.

Thus, to illustrate how POSAR test operates and to check its advantage over AFC, the following twelve test cases have been sampled. Out of these twelve, eleven are real life situations while one is a fabricated test case (done in the lab). The real life situations should help to illustrate how effective POSAR is over AFC. The fabricated case would illustrate how POSAR too, like AFC, will accurately predict the fallibility of comparison as a method when the non-compatible nature of the test materials essentially renders them non-comparable.

8.4.1 Test Case 1: This is a forensic comparison of two data base tables (of banking software), one from the original software (Table 8.2) and another from the pirated software (Table 8.3). These two database tables have been taken from a real life case, Sesame Software Solutions Private Limited V Perfect Software Solutions Private Limited. The nature of the crime is as follows. Sesame Software Solutions Private Limited, the developer of a Co-operative banking Software, had filed a suit in the District Court, Kozhikode, alleging that Perfect Software Solutions Private Limited (a company formed by the ex-employees of the plaintiff company) had committed copyright violation, infringement and software piracy. The computer evidence and material objects involved in this case were collected through a raid by the Advocate Commissioner and the Expert Commissioner (this researcher) appointed by the court. Subsequently, the seizure report along with the computer evidence and material objects were submitted to the court. The court later appointed the RCCF to analyse the computer evidence and material objects and to present their expert report on the suspected copyright violation, infringement and software piracy. The RCCF had submitted an interim report in 2008 and the final report in 2010. For comparing the two software packages,

---

17 Crime number OS 2/07 of District Court, Kozhikode, Kerala, India
19 Resource Centre for Cyber Forensics of Centre for Development of Advanced Computing (CDAC, a Government of India organisation), Thiruvananthapuram, Kerala, India
the RCCF had used a different tool than AFC and POSAR and so, the results in
the RCCF report are irrelevant for and out of the scope of this thesis. However,
the following two database tables are taken from this case and compared using
AFC and POSAR in order to verify the advantage of POSAR over AFC. The
result of this forensic comparison is presented below.

CREATE TABLE dbo.GLMarketValue
(
GlDate datetime NULL ,
GlValue float NULL ,
PrintThisBond tinyint NULL ,
BondPrinter varchar (30) NULL
)
Table 8.2: A data base table from the original software for test case -1

CREATE TABLE [dbo].[GlMarketVal]
(
[GlMarId] [smallint] NOT NULL ,
[GlMarDate] [smalldatetime] NOT NULL ,
[GlMarValue] [smallmoney] NOT NULL ,
[GlMarPrintThisBond] [bit] NOT NULL ,
[GlMarBondWithToken] [bit] NOT NULL ,
[UsrCode] [varchar] (5)  NOT NULL ,
[UsrEnteredOn] [datetime] NOT NULL
) ON [PRIMARY]
Table 8.3: A data base table from the pirated software for test case -1

8.4.1.1 Result of the forensics of test case -1 using POSAR: The two data base tables
were compared using POSAR and the final evidence was available in the form of
sets SP1, SP2, SP3, SO1, SO2, SO3 and the two lists and the report format (Table
8.1), as explained above. The following points emerged out of the forensic
comparison. For simplicity and for judicial convenience, only the final and the
relevant points in the results are presented below.

8.4.1.1.1 Evidence-1: An unnecessary / strange field name called ‘PrintThisBond’
appears on both these tables. In the original software it appeared as just
‘PrintThisBond’ while in the Pirated software, it is appeared as GlMarPrintThisBond. Because the term PrintThisBond can be an acronym of “Printing of Bond” or “Whether the Bond has been Printed or not?” etc., this term and hence the entire field name ‘PrintThisBond’ is generic. Such a generic field has not been used anywhere else in the original software and so, this field, according to the definition (see sections 6.1 and 6.2), is a programming blunder (even if generic) in the original software.

This programming blunder appears identically in respondent’s table too in the form of ‘GlMarPrintThisBond’. As ‘GlMar’ is general thumb impression of most fields in the respective data table of the respondent, further analysis of this field is required by re-engineering (removing/discardng) this thumb impression. Thus, the last part of this field name is PrintThisBond which has 100% similarity with the ‘PrintThisBond’ that appears in the complainant’s data table. Similarly, the term GlMarPrintThisBond which is present in the respondent’s table above does not appear anywhere else in pirated software, and so, GlMarPrintThisBond is also a programming blunder in the pirated software.

8.4.1.1.2 Evidence-2: In SP2 set of POSAR, the field by name ‘GlMarPrintThisBond’ has been included as a suspected post piracy modified field.

All these points show that POSAR comparison has identified several factors (for instances, similarities, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy.

8.4.1.2 Result of the forensics of test case -1 using AFC: As AFC has no report format, AFC’s results (of test case-1) can be listed as follows.

---

22 A generic field / variable name is the name of any field / variable that has been inherited from the functional area (in this case, banking). For example, in a banking software, the account number of all customers are usually stored in fields having generic names such as ‘AcnO’, ‘Acctno’ etc. A Generic item is generally treated as globally common and hence is not considered for final comparison.
8.4.1.2.1 **No evidence found**: Because all the elements in the code segment are filtered out during the filtration stage of AFC, there was no factor found that might contribute to establishing piracy.

8.4.2 **Test Case 2**: This is a forensic comparison of two code segments (of banking software), one from the original software (Table 8.4) and another from the pirated software (Table 8.5), both taken from the RCCF report mentioned in test case-1. Here also, for comparing the two code segments, the RCCF had used a different tool than AFC and POSAR and so, the results in the RCCF report are irrelevant for and out of the scope of this thesis. However, the following two code segments are taken from this case and compared using AFC and POSAR in order to verify the advantage of POSAR over AFC. The result of this forensic comparison is presented below.
Declare TranForDay CURSOR For  
Select SbTrnAmt,SbTrnType  
from SBTrans  
where SBAcNo=@SbAcno and SbTrnDate=@Intdate  
order by SBVoucherNo  
Open TranForDay  
FETCH Next FROM TranForDay INTO @SbTrnAmt,@SbTrnType  
WHILE (@@fetch_status <> -1) BEGIN  
if @SbTrnType='R'  
Select @Balance = @Balance + @SbTrnAmt  
else if @SbTrnType='P'  
Begin  
Select @Balance = @Balance - @SbTrnAmt  
if @Balance < @Bal  
Select @Bal = @Balance  
End  
FETCH Next FROM TranForDay INTO @SbTrnAmt,@SbTrnType  
END  
Close TranForDay  
DEALLOCATE TranForDay  

Table 8.4: A code segment found in the original software for test case -2  

/* DECLARE curTDate CURSOR FOR  
SELECT   SbtAmount,SbtRecOrPay  
FROM     SBTrans  
WHERE    SbmAccNo=@SbmAccNo AND SbtTypeCode=@SbtCode AND  
SbtDate=@Todate  
ORDER BY VoucherNo  
OPEN curTDate  
FETCH NEXT FROM curTDate INTO @curTrnAmount , @curRecOrPay  
WHILE (@@fetch_status <> -1) BEGIN  
IF @curRecOrPay='R'  
SET @CurBal=@CurBal + @curTrnAmount;  
ELSE  
BEGIN  
SET @CurBal=@CurBal - @curTrnAmount;  
IF @MinBal > @CurBal  
SET @MinBal=@CurBal;  
END  
FETCH NEXT FROM curTDate INTO @curTrnAmount , @curRecOrPay  
END  
CLOSE curTDate  
DEALLOCATE curTDate*/  

Table 8.5: A code segment found in the pirated software for test case -2
8.4.2.1 Result of the forensics of test case -2 using POSAR: The two code segments were compared using POSAR and the following points emerged. For simplicity and for judicial convenience, only the final and the relevant points in the results are presented below and not the entire report.

8.4.2.1.1 Evidence-1: These two code segments look algorithmically somewhat identical. (These two are not exactly identical because not all the variable names used in these two segments are fully identical.)

8.4.2.1.2 Evidence-2: In the pirated software, the entire code segment (in Table 8.5) is blocked while in the original software, the code segment (in Table 8.4) is live.

8.4.2.1.3 Evidence-3: The following two situations of perfect similarity in nomenclature have been found; (a) The variable name fetch_status appears in both; and (b) The name of the data base table is SBTrans in both.

All these points show that POSAR comparison has identified several factors (for instances, similarities, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy.

8.4.2.2 Result of the forensics of test case -2 using AFC: As AFC has no report format, AFC’s results (of test case-2) can be listed as follows.

8.4.2.2.1 No evidence found: Because all the elements in the code segment are filtered out during the filtration stage of AFC, there was no factor found that might contribute to establishing piracy.

8.4.3 Test Case 3: This is a forensic comparison of two data base tables (of Hospital Management software), one from the original software (Table 8.6) and another from the pirated software (Table 8.7). These two database tables have been taken
from a real life case, Software Associates V Three others\textsuperscript{23}. The nature of the crime is as follows. Software Associates, the developer of HIS hospital management software, had filed a complaint with the police, alleging that three of their ex-employees have committed copyright violation, infringement and software piracy. The computer evidence and material objects involved in this case were collected through a police raid\textsuperscript{24}. Subsequently, the police had filed the case in the court. Police had submitted the computer evidence and material objects also in the court. The court later appointed this researcher to analyse the computer evidence and material objects and to present an expert report on the suspected copyright violation, infringement and software piracy. This researcher had submitted the expert report\textsuperscript{25}. For comparing the two software packages, a different tool other than AFC and POSAR was used and so, the results in the expert report are irrelevant for and out of the scope of this thesis. However, the following two database tables are taken from this case and compared using AFC and POSAR in order to verify the advantage of POSAR over AFC.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|}
\hline
Crime number & CMP 10371 / 02 of the court of the Judicial First Class Magistrate IV, Kozhikode, Kerala, India \\
\hline
\end{tabular}
\end{table}

\textsuperscript{23} Crime number CMP 10371 / 02 of the court of the Judicial First Class Magistrate IV, Kozhikode, Kerala, India

\textsuperscript{24} In 2002

\textsuperscript{25} On 18\textsuperscript{th} February, 2003
<table>
<thead>
<tr>
<th>Field</th>
<th>Name</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ACCOUNTHEAD</td>
<td>CHAR(8) NOT NULL</td>
</tr>
<tr>
<td>2</td>
<td>FINYEAR</td>
<td>CHAR(4) NOT NULL</td>
</tr>
<tr>
<td>3</td>
<td>ACCTRANSVOUCHERNUMBER</td>
<td>CHAR(8) NOT NULL</td>
</tr>
<tr>
<td>4</td>
<td>ACCTRANSBILLNUMBER</td>
<td>CHAR(11)</td>
</tr>
<tr>
<td>5</td>
<td>ACCTRANSCHEQUENUMBER</td>
<td>CHAR(10)</td>
</tr>
<tr>
<td>6</td>
<td>ACCTRANSCREDIT</td>
<td>NUMERIC(12,2) DEFAULT 0.00</td>
</tr>
<tr>
<td>7</td>
<td>ACCTRANSDATE</td>
<td>DATE</td>
</tr>
<tr>
<td>8</td>
<td>ACCTRANSDEBIT</td>
<td>NUMERIC(12,2) DEFAULT 0.00</td>
</tr>
<tr>
<td>9</td>
<td>ACCTRANSDESCRIPTION</td>
<td>CHAR(300)</td>
</tr>
<tr>
<td>10</td>
<td>ACCTRANSRECDATE</td>
<td>DATE</td>
</tr>
<tr>
<td>11</td>
<td>ACCTRANSRECONCILE</td>
<td>CHAR(1) DEFAULT 'N'</td>
</tr>
<tr>
<td>12</td>
<td>ACCTRANSTYPE</td>
<td>CHAR(2)</td>
</tr>
<tr>
<td>13</td>
<td>COSTCENTRE</td>
<td>CHAR(2)</td>
</tr>
<tr>
<td>14</td>
<td>DIVISION</td>
<td>CHAR(2)</td>
</tr>
<tr>
<td>15</td>
<td>USERNAME</td>
<td>CHAR(6) NOT NULL</td>
</tr>
<tr>
<td>16</td>
<td>MACHINEID</td>
<td>CHAR(10) NOT NULL</td>
</tr>
</tbody>
</table>

Table 8.6: A data base table from the original software for test case -3

(Bhattathiripad, 2002, p.14)
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Data Type</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ACCOUNTHEAD</td>
<td>CHAR(8)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>2</td>
<td>FINYEAR</td>
<td>CHAR(4)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>3</td>
<td>ACCTRANSCREDIT</td>
<td>CHAR(8)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>4</td>
<td>ACCTRANSBILLNUMBER</td>
<td>CHAR(11)</td>
<td>Default 0.00</td>
</tr>
<tr>
<td>5</td>
<td>ACCTRANSCHEQUENUMBER</td>
<td>CHAR(10)</td>
<td>Default 0.00</td>
</tr>
<tr>
<td>6</td>
<td>ACCTRANSDEBIT</td>
<td>NUMERIC(15,2)</td>
<td>Default 0.00</td>
</tr>
<tr>
<td>7</td>
<td>ACCTRANSDATE</td>
<td>DATE</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>ACCTRANSDESCRIPTION</td>
<td>CHAR(300)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>ACCTRANSISSUEBANKNAME</td>
<td>CHAR(50)</td>
<td>Default 'N'</td>
</tr>
<tr>
<td>10</td>
<td>ACCTRANSRECONCILE</td>
<td>CHAR(1)</td>
<td>Default 'N'</td>
</tr>
<tr>
<td>11</td>
<td>COSTCENTRE</td>
<td>CHAR(2)</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>DIVISION</td>
<td>CHAR(2)</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>VOUCHERTYPE</td>
<td>CHAR(1)</td>
<td>Default 'E'</td>
</tr>
<tr>
<td>14</td>
<td>MODIFIEDUSER</td>
<td>CHAR(6)</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>CREATEDUSER</td>
<td>CHAR(6)</td>
<td>NOT NULL</td>
</tr>
<tr>
<td>16</td>
<td>MACHINEID</td>
<td>CHAR(10)</td>
<td></td>
</tr>
</tbody>
</table>

Table 8.7: A data base table from the pirated software for test case -3 (Bhattathiripad, 2002, p.14)
8.4.3.1 **Result of the forensics of test case -3 using POSAR:** The two data base tables were compared using POSAR and the final evidence was available in the form of sets SP1, SP2, SP3, SO1, SO2, SO3 and the two list and the report format (Table 8.1, above), as explained above. The following points emerged out of the forensic comparison.

8.4.3.1.1 **Evidence in tabular format:** Unlike test case 1 and test case 2, the numerical results of this test case (test case 3) are presented in the prescribed format (see table 8.9, below) because of the large values of suspected piracy percentages appearing in the report. They can also be listed as follows.

1. Table name of the respondent is same as that of the plaintiff (Post-piracy modification suspected)
2. 100% similarity in the length of the table
3. 93% of the plaintiff's field names are appearing in the respondent's table too with identical names (generic, however), data types, lengths
4. 100% of the identical fields in both tables are in the same sequence also.

8.4.3.2 **Result of the forensics of test case – 3 using AFC:** As AFC has no report format, AFC’s results (of test case-3) can be listed as follows.

8.4.3.2.1 **No evidence found:** Because all the elements in the code segment are filtered out during the filtration stage of AFC, there was no factor found that might contribute to establishing piracy
1. **Similarity in the table names**: 100% commonality.
2. **Length of the petitioner’s table**: 19
3. **Length of the respondent’s table**: 19
4. **Percentage of similarity in lengths**: 100% similarity
5. **Field count of the petitioner’s**: 16
6. **Field count of the respondent’s**: 21
7. **Percentage of similarity in field count**: 76.2% similarity
8. **Perfect commonality in the names of fields**: 15 out of 16 names of fields in the original software are found in pirated software also. So, 93% commonality
9. **Perfect commonality in name and data type among fields**: 15 out of 21 fields (in 8 above) have the common name and data types. So, 93% commonality in name and data type.
10. **Perfect commonality in name, data type and length among fields**: 15 out of 16 fields (in 8 above) have same names, data types and length. So, 93% commonality
11. **Perfect commonality in name, data type, length and the default values set in the fields**: 15 out of 16 fields (in 8 above) have same names, data types, length and default values. So, 93% commonality.
12. **Perfect commonality in sequence of the fields with same name**: 15 out of 16 fields with same name, do occur in the same sequence. So, 93% commonality.
13. **Perfect commonality in sequence of the fields with same name, data type, length and default values**: 15 out of 16 fields (with same name, data type, length and default values) do occur in the same sequence. So, 93% commonality.
14. **Count of comparable (mappable) fields including suspected-post-piracy modified / created fields**: 15 out of 21 fields in the respondant’s table can be perfectly or approximately mapped (in terms of names) to at least one field in the petitioner’s table. So, 71 % comparable fields in the respondent’s table.
15. **Count of non-mappable but suspected-post-piracy fields, including ending fields**: 2 out of 21 fields in the respondent’s table could not be properly mapped to any of the fields in the petitioner’s table but they can be suspected to be post-piracy modifications. So, 9.5 % incomparable but suspected fields in the respondent’s table. (So, after re-engineering in the Separation Phase of POSAR (that means, the results of 14 & 15 taken together), the total percentage of the suspected fields has increased from 9.5% to 81%)
16. **Count of non-mappable, non-suspected fields**: 4 out of 21 fields in the respondent’s table could not be properly mapped to any of the fields in the petitioner’s table and do not provide any clue to be suspected as post piracy modification. So, 19 % incomparable, non-suspected fields in the ‘pirated’
17. **Inference**: Piracy is confirmed / largely suspected / loosely suspected / not suspected.

Table -8.8: The result of comparison of two database tables in the test case 3 in the proposed format (post-piracy modifications are also considered. Description of this comparison is given in Chapter 4)
8.4.4 **Test Case 4:** Unlike test cases 1, 2 & 3 above (and 5 to 12 below), the test case-4 was an academically fabricated case of copyright infringement for the purpose of this thesis. This case was fabricated for the purpose of the forensic comparison of two trivial code segments (See Table 8.9 and Table 8.10 below) with different objectives (that means, one for adding two numbers and the other for multiplying two numbers) but both have several other strong similarities, otherwise. In a way, this was a case of wrong comparison (for example, comparing apple with orange) but the fabricated case should illustrate how POSAR too, like AFC, would end up reporting the senseless comparison when a pair of two non-comparable were compared. Also, the objective of such a comparison was to ensure that POSAR behaved correctly even in wrong and trivial forensic situations.

```c
#include <stdio.h>
#include <conio.h>

main()
{
    int a=0, b=0, c=0; /* Declaring the integer variables a & b*/
    clrscr(); /* clearing the screen */
    printf("Give the value of A:"); /* Asking for the value of a*/
    scanf("%d", &a); /* reading the value of a*/
    printf("Give the value of B:"); /* Asking for the value of b*/
    scanf("%d", &b);
    printf("A= %d and B= %d\n", a, b); /* Printing the value of a & b*/
    c=a+b;
    printf("Their sum is %d\n", c);
    getch();
}
```

Table 8.9: A simple C-program for adding two given integers for test case -4
#include <stdio.h>
#include <conio.h>

main()
{
    int a=0,b=0,c=0; /* Declaring the integer variables a & b*/
clrscr(); /* clearing the screen */
printf("Give the value of A:"); /* Asking for the value of a*/
scanf("%d",&a); /* reading the value of a*/
printf("Give the value of B:"); /* Asking for the value of b*/
scanf("%d",&b);
printf("A= %d and B=%d
",a,b); /* Printing the value of a */
c=a*b;
printf("Their product is %d\n",c);
getch();
}

Table 8.10: A simple C-program for multiplying two given integers for test case -4

8.4.4.1 Result of the forensics of test case -4 using POSAR: The two code segments were compared using POSAR and the following points emerged out of the forensic comparison. For simplicity and for judicial convenience, only the final and the relevant points in the results are presented below.

8.4.4.1.1 No evidence found: At the planning stage of POSAR, it was found that these two codes are algorithmically meant for different purposes and have different objectives. At the operationalisation stage of POSAR, all the elements in these two codes are filtered out. POSAR comparison could not identify any factor which should contribute to the decision regarding the establishment of piracy.

8.4.4.2 Result of the forensics of test case -4 using AFC: As AFC has no report format, AFC’s results (of test case-3) can be listed as follows.

8.4.4.2.1 No evidence found: During the abstraction stage of AFC, it was found that these two codes are algorithmically meant for different purposes and have different objectives. During the filtration stage of AFC, all the elements in these two codes are filtered out. So, just like POSAR comparison, above, AFC comparison also could not identify any factor which should contribute to the decision regarding the establishment of piracy.
8.4.5 Test case 5: This is a test case of the comparison of two tables in "Table No. 1" given in Annexure-1, taken out from the expert commissioner report submitted to the honourable court of Judicial I class magistrate, Kozhikode, Kerala, India, case number CMP 10371 / 2002, Software Associates vs. Together Infotech26. The first table belongs to the plaintiff's module while the second, the respondent's module.

8.4.5.1 Result of the forensics of test case -5 using POSAR: The test-5 results are listed below (not in the prescribed format).

1. Table name of the respondent is same as that of the plaintiff (Post-piracy modification suspected)
2. All field names in the respondent table are same as those in plaintiff's table (generic, however)
3. Number of fields (in the table) is exactly same
4. Sequence of appearance of identical fields is also identical
5. 66% identical fields have identical data type and length

All these points show that POSAR comparison has identified several factors which should contribute to the decision regarding the establishment of piracy.

8.4.5.2 Result of the forensics of test case -5 using AFC: As AFC has no report format, AFC’s results (of test case-5) can be listed as follows.

8.4.5.2.1 No evidence found: Because all the elements in the code segment are filtered out during the filtration stage of AFC, there was no factor found that might contribute to establishing piracy.

8.4.6 Test case 6: This is a test case of the comparison of two tables in "Table No. 2" given in Annexure-1. The first table belongs to the plaintiff's module while the second, the respondent's module.

26 This researcher has been the expert commissioner in this case
8.4.6.1 **Result of the forensics of test case -6 using POSAR:** The test-6 results using POSAR are listed below (not in the prescribed format).

1. Table name of the respondent is same as that of the plaintiff (Post-piracy modification suspected)
2. 63% of the plaintiff's field names are appearing in the respondent's table too with identical names (generic, however), data types, lengths
3. 100% of the identical fields in both tables are in the same sequence also
4. A non-generic field "ACCOUNTBEGBALANCE" found in both tables

All these points show that POSAR comparison has identified several factors (for instances, high percentage of identicality, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy.

8.4.6.2 **Result of the forensics of test case -6 using AFC:** As AFC has no report format, AFC’s results (of test case-6) can be listed as follows.

1. A non-generic field "ACCOUNTBEGBALANCE" found in both tables

8.4.7 **Test case 7:** This is a test case of the comparison of two tables in "Table No. 3" given in Annexure-1. The first table belongs to the plaintiff's module while the second, the respondent's module.

8.4.7.1 **Result of the forensics of test case -7 using POSAR:** The test-7 results using POSAR are listed below (not in the prescribed format).

1. A non-generic field "ACCTRANsvOUCHERNUMBER" found in both tables
2. A non-generic field "ACCTRANSBILLNUMBER" found in both tables
3. A non-generic field "ACCTRANSCHEQUENUMBER" found in both tables
4. A non-generic field "ACCTRANSCREDIT" found in both tables.
5. A non-generic field "ACCTRANSDATE" found in both tables
6. A non-generic field "ACCTRANSDEBIT" found in both tables
7. A non-generic field "ACCTRANSDESCRIPTION" found in both tables
8. A non-generic field "ACCTRANSRECDATE" found in both tables
9. A non-generic field "ACCTRANSRECONCILE" found in both tables
10. A non-generic field "ACCTRANSTYPE" found in both tables
11. Table name of the respondent is same as that of the plaintiff (Post-piracy modification suspected)
12. 93% of the plaintiff's field names are appearing in the respondent's table too with identical names (generic, however), data types, lengths
13. 100% of the identical fields in both tables are in the same sequence also

All these points show that POSAR comparison has identified several factors (for instances, high percentage of identicality, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy.

8.4.7.2 **Result of the forensics of test case - 7 using AFC:** As AFC has no report format, AFC’s results (of test case-7) can be listed as follows.

1. A non-generic field "ACCTRANSVOUCHERNUMBER" found in both tables
2. A non-generic field "ACCTRANSBILLNUMBER" found in both tables
3. A non-generic field "ACCTRANSCHEQUENUMBER" found in both tables
4. A non-generic field "ACCTRANSREDIT" found in both tables.
5. A non-generic field "ACCTRANSDATE" found in both tables
6. A non-generic field "ACCTRANSDEBIT" found in both tables
7. A non-generic field "ACCTRANSDESCRIPTION" found in both tables
8. A non-generic field "ACCTRANSRECDATE" found in both tables

9. A non-generic field "ACCTRANSRECONCILE" found in both tables

10. A non-generic field "ACCTRANSTYPE" found in both tables

8.4.8 **Test case 8:** This is a test case of the comparison of two tables in "Table No. 4" given in Annexure-1. The first table belongs to the plaintiff's module while the second, the respondent's module.

8.4.8.1 **Result of the forensics of test case -8 using POSAR:** The test-8 results using POSAR are listed below (not in the prescribed format).

1. A non-generic field "PREFIXRECEIPT" found in both tables

2. A non-generic field "PREFIXPAYMENT" found in both tables

3. 81% of the plaintiff's field names are appearing in the respondent's table too with identical names (generic, however), data types, lengths

4. 100% of the identical fields in both tables are in the same sequence also.

5. Table name of the respondent is same as that of the plaintiff (Post-piracy modification suspected)

All these points show that POSAR comparison has identified several factors (for instances, high percentage of identicality, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy.

8.4.8.2 **Result of the forensics of test case -8 using AFC:** As AFC has no report format, AFC’s results (of test case-8) can be listed as follows.

1. A non-generic field "PREFIXRECEIPT" found in both tables

2. A non-generic field "PREFIXPAYMENT" found in both tables
8.4.9 **Test case 9:** This is a test case of comparison of procedures 1.1 and 1.2 in the Annexure.-1. The first table belongs to the plaintiff's module while the second, the respondent's module.

8.4.9.1 **Result of the forensics of test case -9 using POSAR:** The test-9 results using POSAR are listed below (not in the prescribed format).

1. Thumb impression level identicality found in the two procedure names (Post-piracy modification suspected)
2. 100% of the plaintiff's field names are appearing in the respondent's procedure too with identical names (generic, however), and data types
3. 100% of the identical fields in both procedures are in the same sequence also
4. The only statement (the INSERT statement) in the plaintiff's procedure has been identically found in respondent's procedure too. So, 100% statement-level similarity found in these two procedures.

All these points show that POSAR comparison has identified several factors (for instances, high percentage of identicality, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy

8.4.9.2 **Result of the forensics of test case -9 using AFC:** As AFC has no report format, AFC’s results (of test case-9) can be listed as follows.

1. The only statement (the INSERT statement) in the plaintiff's procedure has been identically found in respondent's procedure too.

8.4.10 **Test case 10:** This is a test case of comparison of procedures 2.1 and 2.2 in the Annexure.-1. The first table belongs to the plaintiff's module while the second, the respondent's module.

8.4.10.1 **Result of the forensics of test case -10 using POSAR:** The test-10 results using POSAR are listed below (not in the prescribed format).
1. Thumb impression level identicality found in the two procedure names 
   (Post-piracy modification suspected)

2. High degree of similarity in the IF ACCTRANSBILLNUMBER......
   =NULL; statement.

3. High degree of similarity in the IF (CHEQUE......=NULL; statement

4. High degree of similarity in the INSERT statement

5. 84% of the plaintiff's field names are appearing in the respondent's 
   procedure too with identical names (generic, however), and data types.

6. 100% of the identical fields in both procedure are in the same sequence 
   also.

7. 100% statement-level similarity in these two procedures

8. Thumb impression level identicality found in the two procedure names 
   (Post-piracy modification suspected)

All these points show that POSAR comparison has identified several factors 
(for instances, high percentage of identicality, suspected post-piracy 
modifications etc.) which should contribute to the decision regarding the 
establishment of piracy.

8.4.10.2 **Result of the forensics of test case -10 using AFC:** As AFC has no 
report format, AFC’s results (of test case-10) can be listed as follows.

1. The only statement (the INSERT statement) in the plaintiff’s procedure 
   has been identically found in respondent's procedure too.

8.4.11 **Test case 11:** This is a test case of comparison of procedures 3.1 and 3.2 in the 
Annexure.-1. The first table belongs to the plaintiff's module while the second, the 
respondent's module.

8.4.11.1 **Result of the forensics of test case -11 using POSAR:** The test-11 results 
using POSAR are listed below (not in the prescribed format).
1. Thumb impression level identicality found in the two procedure names (Post-piracy modification suspected)

2. High degree of similarity in the SELECT ____ FROM BILLMASTER WHERE ____; statement.

3. High degree of similarity in the EXECUTE statement.

4. High degree of similarity in the INSERT statement.

5. High degree of similarity in the UPDATE statement.

6. High degree of similarity in the DELETE statement.

All these points show that POSAR comparison has identified several factors (for instances, high percentage of identicality, suspected post-piracy modifications etc.) which should contribute to the decision regarding the establishment of piracy.

8.4.11.2 **Result of the forensics of test case -11 using AFC:** As AFC has no report format, AFC’s results (of test case-11) can be listed as follows.

1. High degree of similarity in the SELECT ____ FROM BILLMASTER WHERE ____; statement.

2. High degree of similarity in the EXECUTE statement.

3. High degree of similarity in the INSERT statement.

4. High degree of similarity in the UPDATE statement.

5. High degree of similarity in the DELETE statement.

8.4.12 **Test case 12:** This is a test case of comparison of procedures 4.1 and 4.2 in the Annexure.-1. The first table belongs to the plaintiff's module while the second, the respondent's module.

8.4.12.1 **Result of the forensics of test case -12 using POSAR:** The test-12 results using POSAR are listed below (not in the prescribed format).
1. Thumb impression level identically found in the two procedure names (Post-piracy modification suspected)

2. 100% of the plaintiff's parameters are appearing in the respondent's procedure too with identical names (generic, however), and data types.

3. 94% of the respondent's field names are appearing in the plaintiff's procedure too with identical names (generic, however), data types, lengths.

4. 100% of the identical fields in both procedure are in the same sequence also.

5. The comment /* IF THE DATE….*/ is identically appearing in both procedures.

6. The comment /* GET THE….*/ is identically appearing in both procedures.

7. The comment /* TO GET….*/ is identically appearing in both procedures.

8. The comment /* IT IS YEAR….*/ is identically appearing in both procedures.

9. High degree of similarity in the SELECT statement.

10. High degree of similarity in the XDATE= statement.

11. High degree of similarity in the IF (BILLDATE…..; statement.

All these points show that POSAR comparison has identified several factors which should contribute to the decision regarding the establishment of piracy.

8.4.12.2 Result of the forensics of test case -12 using AFC: As AFC has no report format, AFC’s results (of test case-11) can be listed as follows.

1. The comment /* IF THE DATE….*/ is identically appearing in both procedures.

2. The comment /* GET THE….*/ is identically appearing in both procedures.
3. The comment /* TO GET….*/ is identically appearing in both procedures.

4. The comment /* IT IS YEAR….*/ is identically appearing in both procedures.

5. High degree of similarity in the SELECT statement.

6. High degree of similarity in the XDATE= statement.

7. High degree of similarity in the IF (BILLDATE…..; statement.