CMR TECHNICAL CAMPUS

(Sponsored by CMR Technical Education Society)

UGC AUTONOMOUS

Accredited by NBA & NAAC with 'A' Grade, Approved by AICTE, New Delhi and JNTU, Hyderabad

Kandlakoya (V), Medchal Road, Hyderabad - 501 401

ASSIGNMENT ANSWER BOOKLET

Hall ticket No. : 8 1 7 R 1 A 6 7 J 9

Name of the student: S. Puja

Course : I / II / III / IV - B.Tech. - I / II Sem

Subject Name : Software Testing Methodlogies

Date of Exam :

MARKS AWARDED

Q.No.	a	b	Total
1			
2			
3			
4			
5			
6			
Total			

Signature of the Chief Superintendent	Signature of the Student with Date	Signature of the Invigilator with Date	Signature of the Evaluator
Ameddy	Puja		

Start Writing from Here

1A) Discuss transaction flow testing techniques

Transaction flow testing techniques

1) Inspections, Testing, Reviews

2) Design Tests for $C_1 + C_2$ coverage

3) Additional coverage ($> C_1 + C_2$)

4) Design Testcases for splits & merges

5) Publish Seleded Test paths easly.

6) Buyer's Acceptance - functional & acceptance tests

2. Path Selection

1) Covering Set ($C_1 + C_2$) functionality sensible T+

2) Add difficult paths

Review with designers and implementers

3. Sensitization - path sensitization

1) Functionally sensible paths - simple

2) Errors, exception, external protocol interface paths - Difficult Testing Tx - flows with external interfaces.

4. Instrumentation - path Instrumentation.

1) Link counters are not useful

2) Need
   - Trace
   - Qurxier or which Tokens resided
   - A Running log

5. Test Data Bases:

1. Design & Maintainance of a Test database - Effort

2. Mistakes

a) Unawareness about design of a centrally administered test DB

b) Test DB design by Testers

c) Using one DB for all tests.

6. Test Execution:

1. Use test execution Automation
2. Have to do a large # of tests for C_1+C_2 coverage.

1 B) Write the Applications of data flow Testing

Applications of Data Flow Testing

1) Comparison Random Testing, P_2, Au - by Autofas

a) Au detects more bugs than

b) P_2 with more testcases

c) P_3 with less # of testcases

2) Comparism of # test cases for ACU, APU, AU & ADUP

a) by Weyker using ASSET Testing system

Testcases Normalised $t = a + b * d$

At Most $d+1$ Testcases for P_2

Test cases / Decision

ADUP > AU > APU > ACU > revised - APU

3) Comparison of # test cases for ACU, APU, AU

→ ADUP by Shimeall & levenson

Testcases Normalised $t = a + b * d$

At Most $d+1$ Test cases for P_2

CMR GROUP OF INSTITUTIONS

Test cases / Decision

ADUP - 1/2 APU

AP - AC

2A) Explain in detail about path products and path Expression.

path products -

1) The name of path that consists of two succesive path segments is conveniently supressed by concalenation of path product of the segment names;

2) For example if x and y are defined as x = abcde , y = fghij, then the path corresponding to x followed by y is denoted by xy = abcde fghij

3) Similarly, yx = fghijabcde

ax = aabcde

xa = abcdea

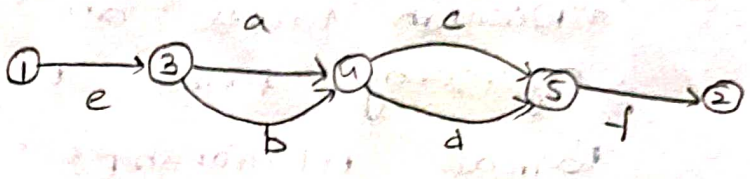
xax = abcdeaabcde

For example,

$X = abc + def + ghi$

$Y = uvw + z$

Then



→ eacf, eadf, ebcf, ebdf

$XY = abcuvw + defuvw + ghiuvw + ubcz + defz + ghiz$

If a link or segment name is repeated, that fact is denoted by an exponent.

Path Expression:

1) Consider a pair of nodes in a graph and the set of paths between those nodes

2) Denote that set of paths by uppercase letter such as X, Y. The members of the path set can be listed as follows

ac, abc, abbc, abbbc, abbbb ....

3) Alteratively, the same set of paths can be denoted by

$ac + abc + abbc + abbbc + abbbbc + \ldots$

4) The + sign is understood to mean "Or" between the two modes of interest, path sac or abc, or abbc, and so one can be taken

2B) What are decision tables? Explain with an example.

a) Decision tables are used in various engineering fields to represent complex logical relationships. This testing is a very effective tool in testing the software.

b) The output may be dependent on many input conditions and decisions table give a tabular views of various combination of input conditions and these conditions are in the form of True (T) and False (F)

Parts of Decision Tables:

The decision table has 4 parts which are divided into portions. They are

1) Condition stubs.
2) Action stubs
3) Condition Entries
4) Action Entries.

Example of Decision table.

	Rule1	Rule2	Rule3	Rule4
Condition 1	Yes	Yes	NO	NO
Condition 2	yes	1	NO	1
Condition 3	NO	yes	NO	1
Action 1	Yes	Yes	no	no
Action 2	No	No	Yes	no
Action 3	NO	NO	NO	yes

Types of Decision Tables:

The decision tables are categorized into two types:

1) Limited Entry 2) Extended Entry

Advantages of Decision Table:

1) The representation is simple so that it can the easily interrupted and is used for development and bussiness as well

2) This table will help to make effective combinations.

Disadvantages of Decision Tree:

1) The main disadvantage is that when the numbers of inputs increases the table will become more complex.

---

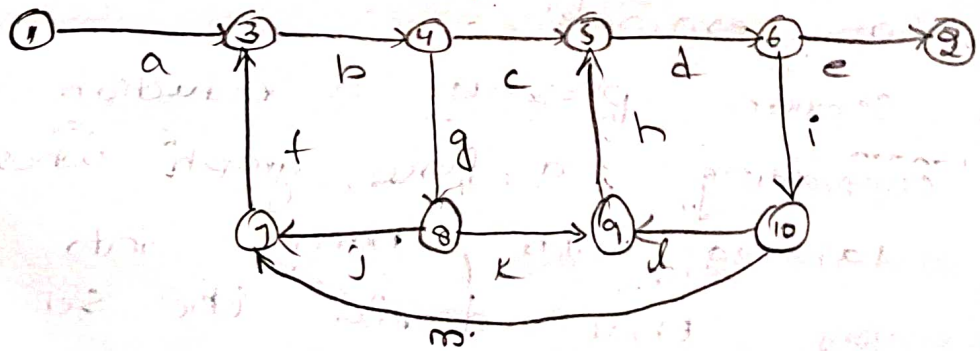
3) Describe reduction procedure algorithm with an example.

This section presents a reduction procedure for converting a flow graph whose links are labeled with names into a path expressions that denotes the set of all entry / exit paths in flow graph

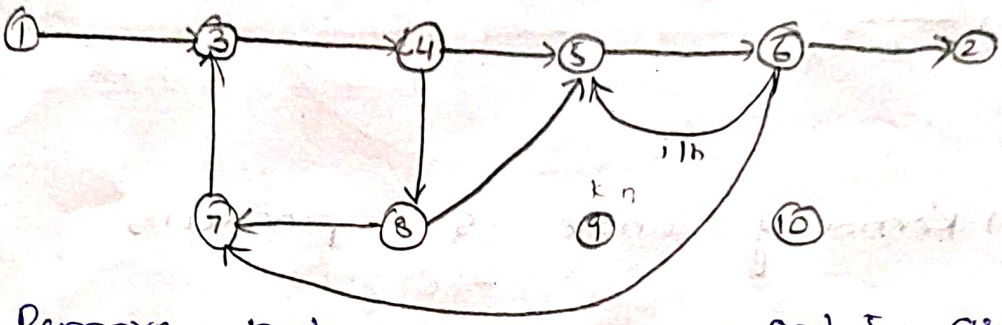
The steps in reduction algorithm are as follows:

1) combine all serial links by multiplying their path expressions.
2) combine all serial parallels links by adding their path expressions.
3) Remove all self-loops by replacing them with a link of the form x^+, where x is the path expression of the link in that loop.
4) Select any node for removal other than the initial or final node.
5) combined any remaining serial links by multiplying their path expressions.
6) Combine any remaining sell parallel links by address their path expressions.
7) Remove all self-loops as in step-3.

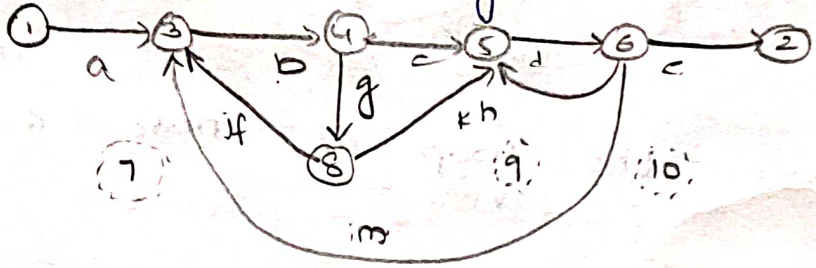
Example:



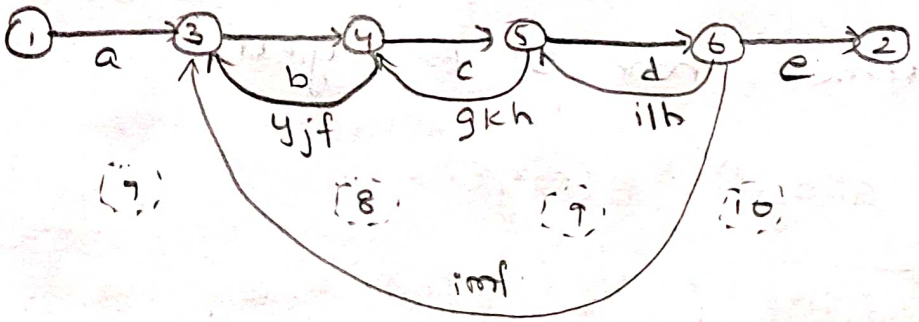
1) Remove node 9 by applying step 4 and 5 to yield.

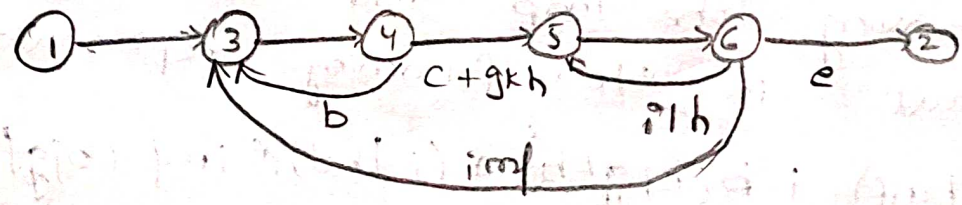


2) Remove node 7 by step 4 and 5, as follows.

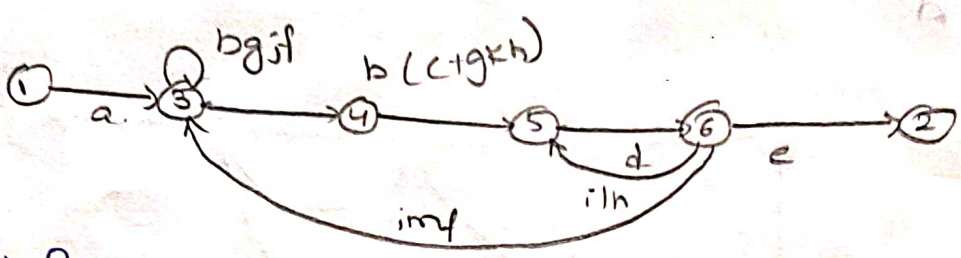


3) Remove node 8 by stop 4 and 5 to obtain

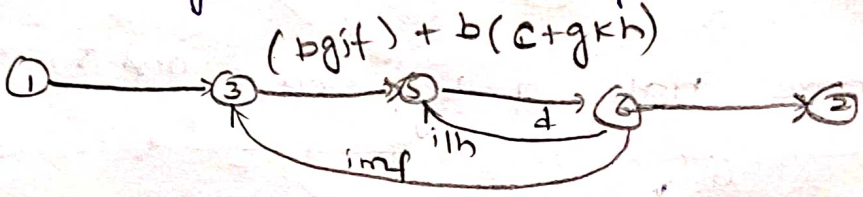




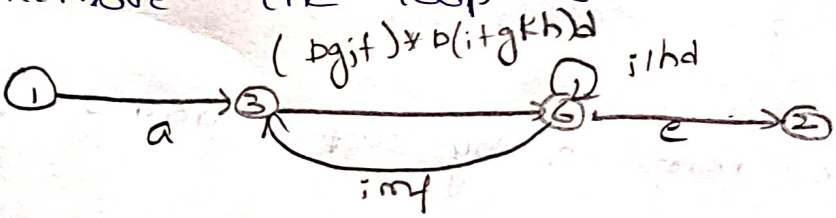
4) Remove node 4 leads to a loop form



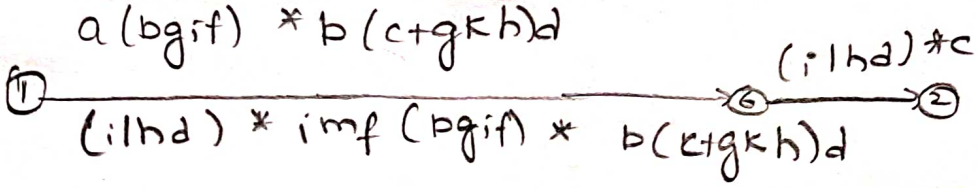
5) Removing node 5 procedures



6) Remove the loop at node 6 to yield



7) Remove node 3 to yield



Removing the loop and then node 6 result in the following representation.

a(bgif) + b(c+gkh) d (ilhd)* imf (bjgf) * b(c+gkh)d* (ilhd)*c

4) Explain about good and bad state graphs in detail.

Good state graph:

1) A state graph is said to be good, when every state, input, transition and output is specified clearly and understandable

2) In good state graph the sequence of inputs is specified for every state in order to perform some action and that will help the system to get back to initial state.

3) It has exactly one transition one specified for every state and input combination so that the transition bugs may not occur.

4) Here only one output action is specified for every transition.

5) In good state graph the bugs are less and easy to identify.

Bad state graph:

1) A state graph is said to be bad, when every state, input transition and output is not specified clearly and

difficult to understand.

2) In bad state graph, there is no sequence of inputs specified, and this result to be the incorrect output.

3) There might be either none or more than one transition specified for every state input combination, and this causes the transition bugs to take place.

4) In bad state graphs the bugs are more and difficult to identify.

---

5) Write a short case study on WinRunner testing tool

WinRunner developed by Mercury interactive, was a pioneering automated functional GUI testing tool used extensively in the Software testing landscape. Its ability to record and replay user interactions, combined with its scripting capabilities, made.

il a popular choice for regression and functional testing.

Implementation of WinRunner:

1) Identification of Testing Needs:
Regression testing to ensure new updates didn't break existing functionality

2) Tool Selection:
- Robust Scripting language (Test script languages)
- Ability to handling the complex, GUI of their financial applications.

3) Setup and Training
- Test engineers were trained on TSL and WinRunner's feature.
- A dedication environment was set up for running automated testing

4) Scripting and Execution:
Test cases were recorded using WinRunner record and playback feature.

5) Continuous Improvement

Regular updates and maintainence of test scripts ensured they remained relevant with each new software relase.