

IDENTIFYING TEST CASES FOR STORED PROCEDURE USING CYCLOMATIC COMPLEXITY APPROACH

ROSMAH BINTI ISMAIL

**UNIVERSITI UTARA MALAYSIA
2009**

QA
36.36
T4S
2801
2009

IDENTIFYING TEST CASES FOR STORED PROCEDURE USING CYCLOMATIC COMPELXITY APPROACH

A dissertation submitted in partial fulfillment of the requirements for the award of the
degree of Master of Science (Information Technology) in the College of Arts and Sciences,
Universiti Utara Malaysia

by

ROSMAH BINTI ISMAIL



KOLEJ SASTERA DAN SAINS
(College of Arts and Sciences)
Universiti Utara Malaysia

PERAKUAN KERJA KERTAS PROJEK
(Certificate of Project Paper)

Saya, yang bertandatangan, memperakukan bahawa
(I, the undersigned, certify that)

ROSMAH ISMAIL
(88348)

calon untuk Ijazah
(candidate for the degree of) **MSc. (Information Technology)**

telah mengemukakan kertas projek yang bertajuk
(has presented his/her project paper of the following title)

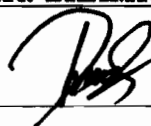
IDENTIFYING TEST CASES FOR STORED PROCEDURE
USING CYCLOMATIC COMPLEXITY APPROACH

seperti yang tercatat di muka surat tajuk dan kulit kertas projek
(as it appears on the title page and front cover of project paper)

bahawa kertas projek tersebut boleh diterima dari segi bentuk serta kandungan
dan meliputi bidang ilmu dengan memuaskan.
(that the project paper acceptable in form and content, and that a satisfactory
knowledge of the field is covered by the project paper).

Nama Penyelia Utama
(Name of Main Supervisor): **MR. ZHAMRI CHE ANI**

Tandatangan
(Signature)

: 

Tarikh
(Date)

: _____

PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for the postgraduate degree from Universiti Utara Malaysia, I agree that University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or part, for scholarly purpose may be granted by my supervisor or, in their absence by the Dean of the Graduate School. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Request for permission to copy or make other use of material in this thesis, in whole or in part, should be addressed to:

**Division of Applied Science
College of Arts and Sciences
Universiti Utara Malaysia
06010 Sintok, Kedah**

ABSTRACT

Testing is very important but expensive and time consuming process in software development. Software quality enhancement can be achieved through the use of a systematic program testing methodology. Few testing methodologies were discussed in this paper such as cyclomatic complexity, specification-based testing, and structural-based automatic program testing.

This paper complements a research on cyclomatic complexity approach in determining the number of test cases that are necessary to achieve thorough test coverage of a particular Oracle stored procedures. Cyclomatic complexity which was developed by Thomas J. McCabe Sr. in 1976 is used to measure the complexity of a program. It directly measures the number of linearly independent paths through a program's source code. The goal of this study is to identify the minimum number of test cases required for a given stored procedure, and cyclomatic complexity was chosen as an approach to achieve this goal. It was discussed thoroughly in chapter 3 of this paper, and testing of this approach was explained in chapter 4. It was found that the higher cyclomatic complexity measure indicates the higher complexity of a program and is more complex to maintain.

ACKNOWLEDGEMENTS

Alhamdulillah, praise to Allah S.W.T whom granted me the strength and ability for the completion of this project.

I am as ever, especially indebted to my loving husband Muhamad Sukri bin Abdul Karim for his love and support throughout my life. His moral support and continuous guidance enabled me to complete my work successfully. I also wish to thank my two daughters Nur Suhailah and Nur Syuhada for their understanding throughout my study.

I would like to offer my sincerest gratitude to my supervisor, Mr Zhamri Che Ani who not only served as my supervisor but also offered valuable advice throughout completion of this project. He patiently guided me through the dissertation process, never accepting less than my best efforts. I thank him very much.

In my daily work I have been blessed with a friendly and cheerful group of fellow colleagues. Many thanks for their contribution of inspirations, joys and supports.

Finally, thanks to Allah for granting me the time and knowledge to complete this study.

TABLE OF CONTENTS

PERMISSION OF USE

ABSTRACT	i
ACKNOWLEDGEMENT.....	ii
LIST OF TABLES	iii
LIST OF FIGURES	iv

CHAPTER 1: BACKGROUND OF THE STUDY

1.1	Introduction	1
1.2	Testing a Stored Procedure in Oracle	3
1.3	Problem Statement	4
1.4	Problem Background	5
1.5	Project Objectives	6
1.6	Project Scope	6
1.7	Significance of the Study.....	6
1.8	Organization of the Report	7
1.9	Summary	7

CHAPTER 2 : LITERARURE REVIEW

2.1	Introduction	8
2.2	Software Testing	9
2.3	The Challenge of Testing	11
2.4	Types of Software Testing	12
2.5	Other Testing Technique	13
2.5.1	Specification-Based Testing.....	13
2.5.2	Automated Testing of Aspect-Oriented Program	14
2.5.3	Structural Testing	14

2.6	Test Cases	18
2.7	Stored Procedures	20
2.7.1	Microsoft SQL Server Stored Procedures	21
2.7.2	Oracle Stored Procedures	22
2.8	Testing a Stored Procedure	24
2.8.1	Automate Testing of Stored Procedures	24
2.8.2	Testing SQL Stored Procedures Using LINQ	26
2.8.3	Other Approaches of Testing Stored Procedures	29
2.8.4	In-Database Stored Procedure Tests	30
2.9	Cyclomatic Complexity	32
2.10	Summary	33

CHAPTER 3 : RESEARCH METHODOLOGY

3.1	Approach of the Study	34
3.2	Comparison of Methodologies Applied in Generating Test Cases	35
3.2.1	Automated Generation of Testcase Datasets	35
3.2.1.1	The Basic of Testing	36
3.2.1.2	Program Structure	39
3.2.1.3	Testing Within the Iteration Structure	44
3.2.1.4	Backtracking	44
3.2.2	Structural-Based Automatic Program Testing	47
3.2.2.1	Test Criteria	47
3.2.2.2	Test Procedures	49
3.2.3	Cyclomatic Complexity	52
3.2.3.1	Key Concept of Cyclomatic Complexity	54
3.3	Identifying Suitable Approach	55
3.4	Applying Cyclomatic Number to Measure Program Complexity	57
3.5	Summary	58

CHAPTER 4 : FINDINGS AND RESULTS

4.1	Result from the Technique used	59
4.2	Java and Oracle Keywords	60
4.3	Testing Oracle Stored Procedures	61
4.4	Results	73
4.5	Summary	73

CHAPTER 5 : CONCLUSION

5.1	Project Summary	74
5.2	Problem and Limitation	74
5.3	Recommendation for Future Project	75
5.4	Conclusion	75

REFERENCES	76
-------------------------	----

LIST OF TABLES

Table 1 :	Oracle Keywords used for Testing	60
Table 2 :	Differences Between Java and Oracle Keywords	61
Table 3 :	Test Cases for check-digit Stored Procedure	63
Table 4 :	Test Cases for string_input Stored Procedure	66
Table 5 :	Test Cases for IF..THEN..ELSIF	68
Table 6 :	Test Cases for Stored Procedure insert_item	70
Table 7 :	Test cases for CASE-END CASE	72
Table 8 :	Summary of CC value	73

LIST OF FIGURES

Figure 1.1 :	Program Testing	2
Figure 1.2 :	PL/SQL Stored Procedure	4
Figure 2.1 :	InsertionSort Function	16
Figure 2.2 :	Control Flow Graph	17
Figure 2.3 :	Paths Derived for InsertionSort Function	17
Figure 2.4 :	Designing Test Cases	19
Figure 3.1 :	Objectives and Processes involved in study	35
Figure 3.2 :	Overview of Software Development Process	36
Figure 3.3 :	Detailed Description of Software Validation Process	37
Figure 3.4(a):	Simple Diagram P	40
Figure 3.4(b):	Subschema Hierarchy	40
Figure 3.4(c):	Tree Representation	41
Figure 3.5:	Typical Iteration Structure	43
Figure 3.6:	Iteration Structure Tree	43
Figure 3.7 :	A Program	47
Figure 3.8 :	The Function Plot of $f(x)$	49
Figure 3.9 :	The Program Instrumented with Counters	51
Figure 3.10 :	Possible Test Cases and the Corresponding Counter Values	51
Figure 4.1 :	A Stored Procedure with IF-ELSE statement	61
Figure 4.2 :	Testing the procedure check_digit	63
Figure 4.3 :	A Stored Procedure with IF-ELSE statement and Logical Operator	64
Figure 4.4 :	Testing the Stored Procedure string_input	65
Figure 4.5 :	A Stored Procedure with IF-THEN-ELSIF	66
Figure 4.6 :	Testing the Stored Procedure Calculate_percentage	67
Figure 4.7 :	A Stored Procedure with BASIC LOOP statement	69
Figure 4.8 :	Testing a Stored Procedure insert_item	70
Figure 4.9 :	A Stored Procedure with CASE-END CASE statement	71
Figure 4.10 :	Testing the Stored Procedure CASE-ENDCASE	72

CHAPTER 1

BACKGROUND OF THE STUDY

This chapter presents the overview of the study, problem statement, objectives, scope, and significance of study.

1.1 Introduction

Testing is a very important but expensive and time consuming process in software development. It can consume at least 50% of the total costs involved in developing software (Wee Kheng Leow,2004). Although there has been steady advancement in formal methods for program verification, testing remains the primary method for discovering faults in software systems. Given a computer program, how can we determine whether or not it will do exactly what it is designed to do? This question is not only intellectually challenging, but also of primary importance in practice.

An ideal solution to this problem would be to develop certain techniques that can be used to systematically construct the formal proof (or disproof) of the correctness of a program. As described in a recent survey by Elspas et al. (J.C Huang,1975), there have

The contents of
the thesis is for
internal user
only

REFERENCES

- Ana C.V. De Melo, Kleber S.Xavior P.N 2008, 'Towards Verification and Testing of Java Programs'. Symposium on Applied computing Proceedings of the 2008, Fortaleza, Ceara, Brazil, ACM. New York, pp.730-734.
- Atif M. Menon, Martha E. Pollack, Mary Lo Soffa 1999, ' Using a Goal-driven Approach to Generate Test Cases for GUIs', Proceedings of the 21st international conference on Software engineering, Los Angeles , ACM, New York, pp.257-266.
- Baowen Xu, J. Q., Xiaofang Zhang,Zhongqiang Wu,Lin Chen 2005, ' A Brief Survey of Program Slicing'. ACM SIGSOFT Software Engineering, vol.30, no.2, pp.1-36.
- Beizer, B 1990, Software Testing Techniques, 2nd Edition, Thompson Computer Press, New York.
- Bertolino, A 2007, ' Software Testing Research: Achievements, Challenges, Dreams', International Conference on Software Engineering a 2007 Future of Software Engineering, Italy, IEEE, Washington, pp.85-103.
- Bergadano F 1993, 'Test Case Generation by means of Learning Techniques'. Proceedings of the 1st ACM SIGSOFT symposium on Foundations of software engineering, Los Angeles, ACM, New York, pp.149-162.
- Baresi L 2001, 'On Accurate Automatic Verification of Publish-Subscribe Architectures', Proceedings of the 29th international conference on Software Engineering. Los Angeles, IEEE Computer Society, Washington, pp.199-208.
- Burleson K.D 2006, Oracle Tuning, The Definition References, Oracle in-Focus Series, online, retrieved March 5,2009 from http://www.praetoriate.com/oracle_tips.htm.
- Basili, V. R 1981, ' Evaluating and comparing software metrics in the software engineering laboratory', Proceedings of the 1981 ACM workshop/symposium on Measurement and evaluation of software quality. New York, ACM, New York, pp.95-106,
- Gupta 2001, Measuring the Complexity of OO Systems, online, retrieved March 2,2009 from <http://javaboutique.internet.com/tutorials/metrics/>.
- David Chays, Saikat Dan, Frank, P. G 2000, ' A Framework for Testing Database Applications', Proceedings of the 2000 ACM SIGSOFT international symposium on Software testing and analysis, Oregon,ACM, New York, pp.147 -157.
- David Willmor, S. M. E 2006, ' An Intensional Approach to the Specification of Test Cases for Database Applications', Proceedings of the 28th international conference on Software engineering, Shanghai, ACM, New York, pp.102-111.

- Donald Burleson K 2006, Oracle DBMS Tips, online, retrieved March 2,2009 from http://www.praetorate.com/oracle_tips.htm.
- Elshoff, J.L 1978, ' On the Use of the Cyclomatic Number to Measure Program Complexity', Journal of advertising, vol.13, no.2, pp.29 - 40.
- Huang, J. C 1975, ' An Approach to Program Testing'. Journal of advertising, vol.7, no. 3, pp.113-128.
- James L.Elshoff, M. M 1978, ' On the use of the cyclomatic number to measure program complexity', Journal of advertising, vol.13, no. 3, pp.29-40.
- James McCaffrey D 2006, 'Software testing certification', online, retrieved March 19,2009 from <http://jamesmccaffrey.spaces.live.com/>
- Javier Jesus Gutierrez, M. E 2006, ' An approach to generate test cases from use cases', Proceedings of the 6th international conference on Web engineering, California, ACM, New York, pp.113-114.
- Javed A.Z, Watson G.N 2007, ' Automated Generation of Test Cases Using Model-Driven Architecture'. Proceedings of the 29th International Conference on Software Engineering Workshops, Queensland, IEEE Computer Society, Washington, pp.200-222.
- Jose Carlos Bregieiro Ribeiro 2008, ' A strategy for evaluating feasible and unfeasible test cases for the evolutionary testing of object-oriented software'. Proceedings of the 3rd international workshop on Automation of software test Leipzig, Germany, ACM, New York, pp.85-92.
- Julian Byke 2006, Oracle 11.1 Keywords, online, retrieved March 20,2009 from http://www.juliandyke.com/Internals/Oracle11_1/Keywords.html
- Laski, J. W 1980, ' A hierarchical approach to program testing', Journal of advertising, vol.15, no. 4, pp.77-85.
- Lopes C.V 2005, ' An analysis of modularity in aspect oriented design', Proceedings of the 4th international conference on Aspect-oriented software development, Chicago, ACM, New York, pp.15-66.
- Miller E.F.,Melton J.R 1975, ' Automated Generation of Testcase Datasets', Proceedings of the International Conference on Reliable Software, Los Angeles,ACM. New York, pp.51-58.
- Mccabe, T. J 1976, ' A complexity measure'. Proceedings of the 2nd international conference on Software engineering, San Francisco, IEEE Computer Society Press, Wasington, pp.407-423.

- Myers, G.J 1977, ' An extension to the cyclomatic measure of program complexity', Journal of advertising, vol.12, no.10, pp.61-64.
- Neena Kochar, E.G., Priya Nathan 1999, Introduction to Oracle: SQL/PL/SQL, Jerry Bosnan.
- Ostrand, T.J 1988, ' The Category-Partition Method for Specifying and Generating Functional Tests', journal of advertising, vol.31, pp.676-686.
- Qing Xie A. M. M 2007, ' Designing and Comparing Automated Test', Oracles for GUI-Based Software Applications, Washington, ACM Transactions on Software Engineering and Methodology (TOSEM), New York, vol.16.
- Reifer D 1974, ' Automating formal specification based testing'. In Proceeding Conference on Theory and Practice of Software Development, Italy, ACM, New York, pp.833-847.
- Richard A. Demillo, A. J. O 1993, 'Experimental results from an automatic test case generator'. Journal of advertising. vol.2, no.10, pp.109-127.
- Sullivan J.E 1973, 'Measurement of program complexity by the pair', Journal of advertising, vol.13, no.3, pp.61-64.
- Shyamal Suhana Chandra K.C 2005, ' A Comparison of Java and C#, Journal of Computing Sciences in Colleges, vol.20, no.2, pp.238-254.
- Stwarz, F 2006, Java Complexity Measurement, online, retrieved March 10, 2009 From http://www.leepoint.net/notes/java/principles_and_practices/complexity/complexity_measurement.html
- Tao Xie, J. Z 2007, ' Perspectives on automated testing of aspect-oriented programs'. Proceedings of the 3rd workshop on Testing aspect-oriented programs, New York, ACM, New York, pp.7-12.
- Thomas J McCabe, C. W. B 1989, ' Design Complexity Measurement and Testing', Journal of advertising, vol.32, no.10, pp.1415-1425.
- Watkins J 2001, IT : An Off-the-Shelf Software Testing Handbook, NY USA, 2nd edition, Cambridge University Press, New York.
- Wee Kheng Leow, S. C. K., Yi Sun 2004, ' Automated Generation of Test Programs From Closed Specification of Classes and Test Cases', Proceedings of the 26th International Conference on Software Engineering. New York, IEEE Computer Society, Washington, pp.96-105.