

UNIVERSITI TEKNOLOGI MARA

**A TASK-BASED PROGRAM
COMPREHENSION FRAMEWORK
FOR NOVICES**

AMAL ABDULGHANY QASSEM AL-SHARGABI

Thesis submitted in fulfillment
of the requirements for the degree of
Doctor of Philosophy

Faculty of Computer and Mathematical Sciences

September 2016

AUTHOR'S DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Teknologi MARA. It is original and is the results of my own work, unless otherwise indicated or acknowledged as referenced work. This thesis has not been submitted to any other academic institution or non-academic institution for any degree or qualification.

I, hereby, acknowledge that I have been supplied with the Academic Rules and Regulations for Post Graduate, Universiti Teknologi MARA, regulating the conduct of my study and research.

Name of Student : Amal Abdulghany Qassem Al-Shargabi
Student I.D. No. : 2009318751
Programme : Doctor of Philosophy in Science
(Computer Science) - CS990
Faculty : Computer and Mathematical Sciences
Thesis Title : A Task-Based Program Comprehension Framework
For Novices
Signature of Student :
Date : September 2016

ABSTRACT

Novices face much difficulty in comprehending even small computer programs. A framework is needed to help novices' attain sufficient program comprehension skills. To develop such framework, this research re-examines the three main factors that influence the novices' program comprehension; the programmer, the program code, and the task. Most studies so far focused on the first two especially programmer, with tasks traditionally applied only as a measure for program comprehension. Very few studies have inverted this supplementary role of task and instead examined the task factor itself. There is a research gap on the effects of tasks on novices' program comprehension. Moreover, current program comprehension mental models (PCMMs) have been mostly constructed for expert programmers, which do not match the novices' mental model. To solve these two problems, the research proposes a task-based program comprehension framework for novices. Employing empirical software engineering research design approach, 1) fourteen effective programming tasks for novices have been identified, 2) a novices' PCMM has been constructed, and 3) the effects of the tasks on the PCMM have been investigated. Consequently, the tasks were classified according to the cognitive domain of Revised Bloom's Taxonomy. The tasks in each cognitive category were then ranked based on their effectiveness on novices' program comprehension using a survey of instructors. Subsequently, novices' PCMM with four core abstraction levels, i.e. Statement, Block, Module, and Program, and an extended level, i.e. Domain, was developed, and validated by experts. The effects of eight tasks on the novices' PCMM were investigated through an experiment involving 69 novices, and six tasks were validated through another experiment involving 178 novices in three universities. Both experiments also validated the ranking of these tasks on the novices' PCMM. The first experiment demonstrated that all the tested tasks were able to significantly improve novices' PCMM. The first and second ranked tasks were studied more closely, where the first ranked tasks consistently showed higher improvements than the second ranked tasks in each of the four tested Bloom's categories of Remember, Understand, Analyze, and Create. The second experiment demonstrated that different tasks improved the abstraction levels and the information categories differently. It also indicated that higher cognitive category tasks improve PCMM at higher abstraction levels. The general implication is that the framework can be an effective tool for computing educators to incorporate program comprehension in programming, and thus shift from merely teaching tracing and debugging tasks only. These tasks need to be introduced in stages in the teaching of programming, starting initially from the lower cognitive categories' tasks such as Recall and culminating at the higher cognitive categories' tasks such as Modification in possibly team project assessments. However, these tasks should be applied with taking the consideration of the novices' programming levels and the information categories need to be improved. The key contribution of this thesis is a new developed framework, which includes novices' PCMM, and a set of classified and ranked effective tasks that can improve novices' PCMM.

ACKNOWLEDGEMENT

Firstly and foremost, all praise and thanks are due to my Lord Allah for granting me the health, knowledge, patience, and wisdom to overcome all of the difficulties I have faced during my PhD journey. I would like to express my gratitude and sincere appreciation to my supervisor Assoc. Prof Dr Syed Ahmad Aljunid, who guided me with her dedicated attention, expertise and continued inspiration throughout the entire period of this study, his enthusiasm for this research and for incredible rate of proofreading. It has been an honour to be one of his Ph.D. students. He has taught me, both consciously and unconsciously, how good Empirical experiments are done. I appreciate all of his contributions of time, ideas, and funding to make my Ph.D. experience successful.

In addition, I am very thankful to Assoc. Prof Dr Muthukkaruppan Annamalai (my co-supervisor) he has showed an excellent example in enhancing the research quality. His uncountable suggestions and valuable advice were greatly treasured. Moreover, I express my gratefulness to Prof Abdullah, who has been a source of consultation and guidance throughout my research.

Furthermore, my special thanks are given to all Universiti Teknologi MARA (UiTM) staff for giving me the chance to be a research member. Besides, I would like to thank staff of each of the following universities: Universiti Kebangsaan Malaysia (UKM), Universiti Malaya (UM), Universiti Putra Malaysia (UPM), Universiti Sains Malaysia (USM), Universiti Tun Hussein Onn Malaysia (UTHM), Universiti Teknologi Malaysia (UTM), Universiti Utara Malaysia (UUM), Asia Pacific Institute of Information Technology (APIIT), Multimedia University (MMU), Management and Science University (MSU), and Universiti Industri Selangor (UNISEL) for granting me the permission to conduct the research questionnaire, survey, and experiments.

Special appreciation also goes to Dr Shuhaidah for her remarkable contribution and support. Her effort is noticeable in order for me to attain the objective of the research. I would like to thank Atika for her cooperation and help in the validation part of my research. Likewise, many thanks go to the statisticians that have shared with me worthy knowledge and tips.

My heartfelt gratitude is given to my husband, Abdulbasit for his immense love and encouragement. Plenty of thanks and appreciation is for my parents who raised me with the love of knowledge and supported me in all my pursuits. Their constant prayers are greatly admired. Additionally, I would like to thank my loving son who has shown patience and toleration for the sake of the success of his mum. Unlimited thanks for my sister Lamis who has helped me in every way to accomplish the goals of the research. I am also grateful to my parents in law for their continued encouragement and unlimited support to pursue my graduate studies. Finally, I am deeply grateful to all my brothers, sisters, friends, and to everyone who had a role in making this possible.

TABLE OF CONTENT

	Page
CONFIRMATION BY PANEL OF EXAMINERS	i
AUTHOR'S DECLARATION	ii
ABSTRACT	iii
ACKNOWLEDGEMENT	iv
TABLE OF CONTENT	v
LIST OF TABLES	x
LIST OF FIGURES	xiv
LIST OF ABBREVIATIONS	xviii
CHAPTER ONE: INTRODUCTION	1
1.1 Background	1
1.2 Motivation	3
1.3 Problem Statement	9
1.4 Research Aim	11
1.5 Research Questions	11
1.6 Research Objectives	11
1.7 Overview of Research Design	11
1.8 Research Scope	13
1.9 Significance	14
1.10 Thesis Outline	15
CHAPTER TWO: LITERATURE REVIEW	16
2.1 Introduction	16
2.2 Programming Pedagogy	16
2.2.1 Novices Definition	17
2.2.2 Programming Pedagogy Issues	18
2.3 Program Comprehension	20
2.3.1 The Importance of Program Comprehension for Novices	20
2.3.2 Elements of Program Comprehension	23