

Third Edition

SEPTEMBER 2014

Data Structures and Algorithms

Nor Bahiah Hj. Ahmad
Dayang Norhayati Abang Jawawi
Zalmiyah Zakaria



Faculty of Computing, Universiti Teknologi Malaysia

DATA STRUCTURES AND ALGORITHMS

Nor Bahiah Hj. Ahmad
Dayang Norhayati Abang Jawawi
Zalmiyah Zakaria

Copyright © 2011, 2012, 2014

All rights reserved. No part of this publication may be reproduced or distributed in any form or by any means, or stored in a data base or retrieval system, or transmitted in any form or by any means, electronics, mechanical, photocopying or otherwise, without the prior written permission of the authors.

Disclaimer

This workbook and its contents are intended solely for the use of Data Structures and Algorithms course to be taught at Faculty of Computing, Universiti Teknologi Malaysia, Skudai. The algorithms and programs in this module have been included for their instructional value. They have been tested but are not guaranteed for any particular purpose. The authors do not offer any warranties or representations nor do they accept any liabilities with respect to their use. The authors reserve the right to revise this publication and make changes from time to time in its content without notice.

Third Edition
September 2014

PREFACE

OBJECTIVE

The teaching module for Data Structures and Algorithms is intended to be used as teaching and learning reference as a practical problem-solving class activity.

LEARNING OUTCOMES

By the end of the course, students should be able to:

No.	Course Learning Outcome	Programme Learning Outcome(s) Addressed	Assessment Methods
1.	Describe all types of data structures and know how to apply them effectively in problem solving.	PO1	LE, Q, T, F
2.	Analyze and select the most suitable and effective algorithm for solving certain problem and case study.	PO2	LE, Q, A, T, PR, F, LT
3.	Develop a medium to complex program as a group mini project by applying data structure concept and present the output in the class.	PO2, PO6	PR, A, Pr, Peer

Remarks :

PO1 – Knowledge Understanding (Ability to acquire and apply theory and principles of CS)

PO2 – Technical (Ability to design and construct computer program using standard approaches)

PO6 – Team Working (ability to work effectively in a team)

T – Test, Q – Quiz, LE – Lab exercise, A – Assignment, Peer – Peer assessment, PR – Project, Pr – Presentation, F – Final Exam, LT- Lab Test

CONTENT OF THE MODULE

The module consists of three parts: Key concepts, Programming Exercise and Exercise. The objective of Key Concepts material is to be used for class lecture session. The content of the material is in the form of short notes that explain the main concepts of data structure. Programming exercises provide problem solving activities whereby the students apply the data structure concepts effectively in solving problem in C++. Exercises contains various type of questions in order to asses the students knowledge on data structure concepts theoretically and practically. The learning outcomes or objectives for each module are stated at the beginning of every module. Mini project and assignments in this course are to expand and evaluate problem

solving skills of the students. The specification of the assignment and mini project is included in the Appendix of this module.

The data structure concepts included in this module are: Introduction To Data Structure, Abstract Data Type and Class, Recursive, Algorithm Efficiency, Sorting Techniques, Searching, Link List, Stack, Queue, and Tree.

ASSESSMENT

No.	Assessment	Number and Topics	% each	% total	Theory	Practical
1	Assignments (group project 2 students)	2 1. C++, Sorting and Searching 2. Linked List	4%	8	-	Practical (8%)
2	Quizzes	2-4 1. C++ 2. Recursive 3. Algo Efficiency 4. Sorting 5. Linked List 6. Stack 7. Queue 8. Tree	5%	10	Theory (5%)	Practical (5%) C++/ Recursive
3	Lab Exercises (individual)	4 1. C++ 2. Sorting 3. Linked List 4. Stack/Queue	2.5%	10		Practical (10%)
4	Project (group work)	1 (Cover everything and include either stack/queue/tree)	9%	9		Practical (9%)
	Peer Review	3	1%	3		Practical (3%)
5	Test 1	(Topics in Week1,ADT-Week5(Sorting))	20%	20	Theory (20%)	
6	Final Exam	(Topics in Week6-Week14)	40%	20	Theory (20%)	
7	Test 2 (Lab Test)			20		Practical (20%)
Overall Total				100	45%	55%

ACKNOWLEDGMENT

The authors gratefully acknowledge all contributors to this workbook in term of ideas, selected questions and comments. Thank you to the Software Engineering Department and Faculty of Computing for their support in preparing this material.

REFERENCES

1. Frank M Carano and Timothy Henry, "Data Abstraction and Problem Solving with C++", Walls and Mirrors, Sixth Edition, (2013), Pearson.
2. Nor Bahiah Hj Ahmad and Dayang Norhayati Abang Jawawi. Data Structure and Algorithm Lecture Slides. URL: <http://ocw.utm.my/course/view.php?id=31>. Accessed on September 2014.
3. Nor Bahiah Ahmad et al, "*Struktur Data & Algoritma Menggunakan C++* ", UTM Press, 2005.
4. Richrd F. Gilberg and Behrouz A. Forouzan, "*Data Structures A Pseudocode Approach With C++*", Brooks/Cole Thomson Learning, 2001.

CONTENTS

MODULE	TOPIC	PAGE
1	INTRODUCTION TO DATA STRUCTURE	1
	Objectives For Students	1
	1.0 Software Engineering And Problem Solving	1
	2.0 Algorithm	2
	3.0 Data Structure	3
	4.0 Programming Paradigm	7
	5.0 Programming Principle	8
	Exercises	11
2	ABSTRACT DATA TYPE AND CLASS	15
	Objectives For Students	15
	1.0 Abstract Data Type	15
	2.0 C++ Class	17
	3.0 Class As Function Parameter	23
	4.0 Class As Return Value From Function	26
	5.0 Array Of Class	27
	6.0 Pointer To Object	29
	7.0 Basic C++ File Opertions	31
	8.0 Header File And Implementation File	33
	Programming Exercises	37
	Exercises	48
3	RECURSIVE	55
	Objectives For Students	55
	1.0 Introduction To Recursion	55
	2.0 Designing Recursive Algorithm	56
	3.0 Implementation Of The Recursive Algorithms	56
	Programming Exercises	64
	Exercises	68

4	ALGORITHM EFFICIENCY	73
	Objectives For Students	73
	1.0 Introduction To Algorithms	73
	2.0 Complexity Of Algorithm	74
	Programming Exercises	83
	Exercises	87
5	SORTING	93
	Objectives For Students	93
	1.0 Introduction To Sorting	93
	2.0 Sorting Process	94
	3.0 Sorting Algorithms	95
	4.0 Bubble Sort	96
	5.0 Selection Sort	103
	6.0 Insertion Sort	107
	7.0 Summary Of Quadratic Sorting Algorithms Complexity	111
	8.0 Merge Sort	111
	9.0 Quick Sort	117
	10.0 Summary	123
	Programming Exercises	124
	Exercises	126
6	SEARCHING	135
	Objectives For Students	135
	1.0 Introduction To Searching	135
	2.0 Basic Sequential Search	136
	3.0 Binary Search	140
	Programming Exercises	145
	Exercises	151
7	LINKED LIST	159
	Objectives For Students	159
	1.0 Introduction To Linear List	159
	2.0 Array As A Linear List	160
	3.0 Linked List	163
	4.0 Implementation Of Linked List	165

	Programming Exercises	171
	Exercises	175
8	STACK	179
	Objectives For Students	179
	1.0 Introduction To Stack	179
	2.0 Stack Implementation	181
	3.0 Array Implementation Of Stack	181
	4.0 Linked List Implementation Of Stack	186
	5.0 Stack Application	190
	Programming Exercises	200
	Exercises	206
9	QUEUE	215
	Objectives For Students	215
	1.0 Introduction To Queue	215
	2.0 Queue: Linear Array Implementation	217
	3.0 Queue: Circular Array Implementation	221
	4.0 Queue Implementation Linked List	223
	5.0 Queue Implementation: Linear Linked List And Circular Linked List	225
	Programming Exercises	230
	Exercises	235
10	TREE	243
	Objectives For Students	243
	1.0 Introduction To Tree	243
	2.0 Tree Terminology	245
	3.0 Binary Tree	247
	4.0 Binary Search Tree	252
	5.0 Binary Search Tree Implementation	253
	Programming Exercises	268
	Exercises	270

LIST OF APPENDICES

APPENDIX I	TEST 1 SCSJ 2013, SEMESTER 1 20132014	279
APPENDIX II	FINAL EXAM SCSJ 2013, SEMESTER 1 20132014	291
APPENDIX III	ASSIGNMENTS AND MINI PROJECT SPECIFICATION	301
APPENDIX IV	SAMPLE PROJECT DOCUMENTATION	305
APPENDIX V	PEER REVIEW ASSESSMENT SAMPLE	312