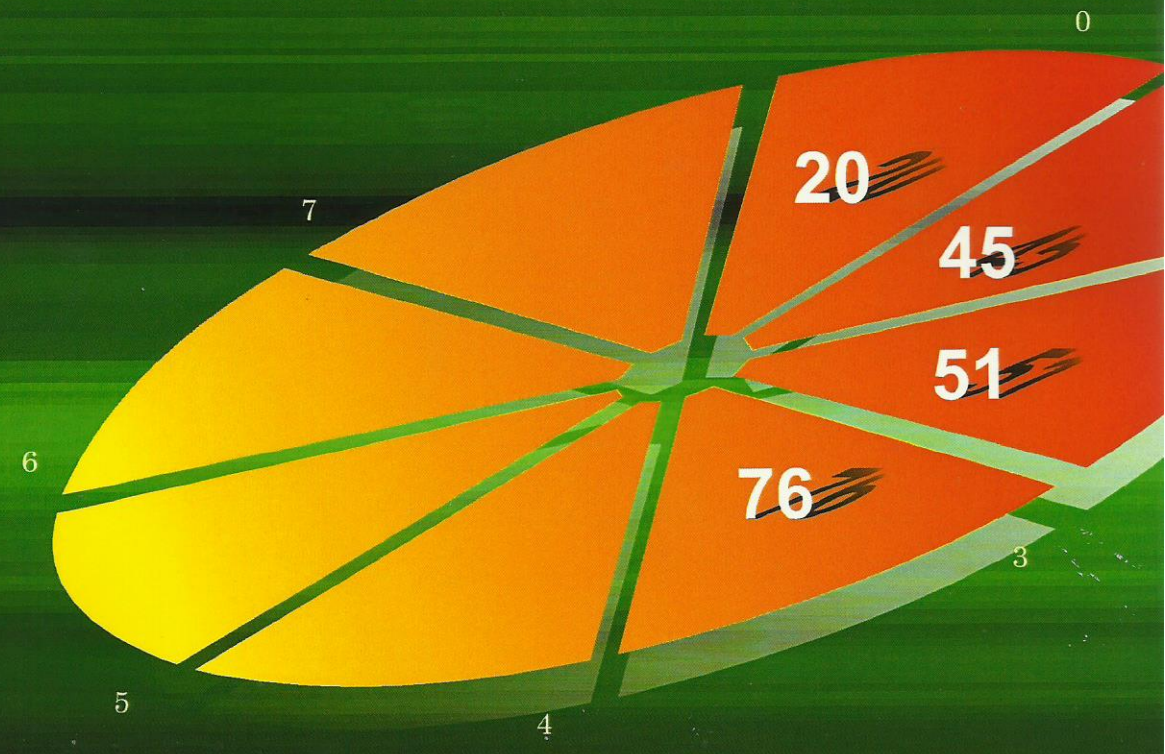
Second Edition

SEPTEMBER 2012

Data Structures and Algorithms



Nor Bahiah Hj. Ahmad
Dayang Norhayati Abang Jawawi
Zalmiyah Zakaria

Universiti Teknologi Malaysia

Fakulti Sains Komputer dan Sistem Maklumat

DESKTOP PUBLISHER

DATA STRUCTURES AND ALGORITHMS

Nor Bahiah Hj. Ahmad

Dayang Norhayati Abang Jawawi

Zalmiyah Zakaria

Copyright © 2012

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 Fakulti Sains Komputer dan Sistem Maklumat, 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.

Second Edition
September 2012

INTRODUCTION

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.

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 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.

ACKNOWLEDGEMENT

The authors gratefully acknowledge all contributors to this workbook in term of ideas, selected questions and comments. Thanks to the Software Engineering Department and Faculty of Computer Science and Information Systems for their support in preparing this material.

REFERENCES

1. Nor Bahiah Ahmad et al, "*Struktur Data & Algoritma Mengunakan C++* ", UTM Press, 2005.

2. Frank M Carano, Janet J Prichard, "Data Abstraction and Problem Solving with C++", Walls and Mirrors, Fifith Edition, (2007), Addison Wesley

3. 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	8
	5.0	Programming Principle	9
	Exercises		11
2	**ABSTRACT DATA TYPE AND CLASS**		**17**
	Objectives For Students		17
	1.0	Abstract Data Type	17
	2.0	C++ Class	19
	3.0	Class As Function Parameter	24
	4.0	Class As Return Value From Function	27
	5.0	Array Of Class	28
	6.0	Pointer To Object	30
	7.0	Basic C++ File Opertions	32
	8.0	Header File And Implementation File	34
	Programming Exercises		37
	Exercises		51
3	**RECURSIVE**		**59**
	Objectives For Students		59
	1.0	Introduction To Recursion	59
	2.0	Designing Recursive Algorithm	60
	3.0	Implementation Of The Recursive Algorithms	60
	Programming Exercises		67
	Exercises		71

4	**ALGORITHM EFFICIENCY**		**75**
	Objectives For Students		75
	1.0	Introduction To Algorithms	75
	2.0	Complexity Of Algorithm	76
	Programming Exercises		83
	Exercises		87
5	**SORTING**		**91**
	Objectives For Students		91
	1.0	Introduction To Sorting	91
	2.0	Sorting Process	93
	3.0	Sorting Algorithms	93
	4.0	Bubble Sort	94
	5.0	Selection Sort	101
	6.0	Insertion Sort	105
	7.0	Summary Of Quadratic Sorting Algorithms Complexity	109
	8.0	Merge Sort	109
	9.0	Quick Sort	115
	10.0	Summary	121
	Programming Exercises		122
	Exercises		125
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		152
7	**LINKED LIST**		**161**
	Objectives For Students		161
	1.0	Introduction To Linear List	161
	2.0	Array As A Linear List	162
	3.0	Linked List	165
	4.0	Implementation Of Linked List	167

Programming Exercises 173
Exercises 179

8 **STACK** **183**

Objectives For Students 183

1.0 Introduction To Stack 183
2.0 Stack Implementation 185
3.0 Array Implementation Of Stack 185
4.0 Linked List Implementation Of Stack 190
5.0 Stack Application 195

Programming Exercises 204
Exercises 210

9 **QUEUE** **219**

Objectives For Students 219

1.0 Introduction To Queue 219
2.0 Queue: Linear Array Implementation 221
3.0 Queue: Circular Array Implementation 225
4.0 Queue Implementation Linked List 227
5.0 Queue Implementation: Linear Linked List And Circular Linked List 229

Programming Exercises 232
Exercises 240

10 **TREE** **249**

Objectives For Students 249

1.0 Introduction To Tree 249
2.0 Tree Terminology 251
3.0 Binary Tree 253
4.0 Binary Search Tree 258
5.0 Binary Search Tree Implementation 259

Programming Exercises 274
Exercises 277

APPENDIX 1 SAMPLE PROJECT DOCUMENTATION **287**