CHIANG MAI UNIVERSITY

Bachelor of Science (Software Engineering)

College of Arts, Media and Technology

2nd Semester / Academic Year 2011

---

1. Course

SE 102 (9533102) Abstract Data Type and Problem Solving 3 (3 – 0 – 6) credits

Prerequisite: SE 101 (953101) COMPUTERS AND PROGRAMMING

Course Description

This course is concerned with abstract data types, information hiding, modularity and implementations using various data structures. The topics also include basic knowledge of problem solving strategies and issues related to static and dynamic implementation strategies.

2. Instructors

Section 701 9:30-11 Tuesday, Friday Room 213 CAMT

Pree Thiengburanathum

pree.t@cmu.ac.th

Office Room 417

Section 702 9:30-11 Tuesday, Friday, Room RB 5402

Parinya Suwansrikham

parinsu@hotmail.com

Office Room 417

3. Course Objectives

After completing this course, students will be able to:

3.1 Explain basic knowledge in abstract data types, information hiding and modularity

3.2 Apply fundamental data structures to represent data for problem solving

3.3 Analyze and design the abstract data types for program design

3.4 Implement techniques of computer-based problem solving using ADTs

4. Course Texts

1.Data Abstraction and Problem Solving with Java, 3rd edition, 2011, Frank Carrano, Janet Prichard

Course Website

Please register and check the course announcement all homework assignments will be post at the following website.

http://cmuonline.cmu.ac.th/

5. Course Contents

Week/Date	Topic	Lecture hour	Course activity	Lecturer
1 November 11, 11	Introduction to the course - Survey and course overview - Basic I//O - Program submission	1.5	Pre-test & discussion	Pree/Parinya
1 November 15, 11	Basic Concepts I - Pseudo code - Algorithm - The ADT	1.5	Lecture & discussion	Pree/Parinya
2 November 18, 11	Basic Concepts II - Algorithm efficiency	1.5	Lecture & discussion	Pree/Parinya
2 November 22, 11	Recursion - Basic of recursion - Recursion example - *Programming Assignment #1*	1.5	Lecture & discussion	Pree/Parinya
3 November 25, 11 November 29, 11	Linear Lists - Basic Operation - Implementation - Applications	3	Lecture & discussion	Pree/Parinya
4 December 4, 11 December 8, 11	Queues - Queue Operations - Queue Linked List Design - Queue Applications	3	Lecture & discussion	Pree/Parinya

5 December 11,11 December 15, 11	Stack - Basic Stack Operations - Stack Linked List Implementation - Stack Applications - *Programming Assignment #2*	3	Lecture & discussion	Pree/Parinya
6 December 18,11	Introduction to Trees - Basic tree Concepts - Binary Trees - General Trees	1.5	Lecture & discussion 1.5 hrs.	Pree/Parinya
6 December 22, 11	Binary Search Trees - Basic Concepts - BST Operations - BST Applications - Review for Midterm Examination	1.5	Lecture & discussion 1.5 hrs.	Pree/Parinya
Midterm Examination (24 -30 December)				
7 January 3, 12 January 6, 12	AVL Trees - AVL Tree Basic Concepts - AVL Tree Implementations - AVL Tree Algorithms - Application	3	Lecture & discussion 3 hrs.	Pree/Parinya
8 January 10, 12 January 13, 12	Heaps - Basic Concepts - Heap Implementation - Heap Applications - *Programming Assignment #3*	3	Lecture & discussion 3 hrs.	Pree/Parinya
Commencement day ** no class**				
9 January 24, 12 January 27, 12	Multi-way Trees - M-way Search Trees - B-trees Implementation	3	Lecture & discussion 3 hrs.	Pree/Parinya
10 February 3, 12 February 7, 12	Graphs - Basic Concepts - Operations	3	Lecture & discussion 3 hrs.	Pree/Parinya

	- Networks			
11,12 February 10, 12 February 14, 12 February 17, 12 February 21, 12	Searching - List Searches - Search Implementations - Hashed List Searches - Collision Resolution - *Programming Assignment #4*	6	Lecture & discussion 6 hrs.	Pree/Parinya
13,14 February 24, 12 February 27, 12	Sorting - Sort Concepts - Selection Sort - Insertion Sorts - Bubble Sort	6	Lecture & discussion 6 hrs.	Pree/Parinya
15 March 2	Review before final exam	1.5	Discussion 1.5 hrs.	Pree/Parinya
Final Examination (28 Feb - 12 March 12)				
Total lecture 45 Hours				

Note: Some topics of the contents might be subject to change or add without notice in advance.

6. Course Requirements

6.1 Lectures in class

6.2 Quizzes and Programming assignments

6.3 Reading assignments

6.4 Paper-based exams

7. Grading System

The semester grade is computed:

7.1 4 x Programming Assignments	*30%*
7.2 Attendance	*5%*
7.3 4 x Quizzes	*5%*
7.4 Midterm Examination	*30%*
7.5 Final Examination	*30%*
Total	100%

8. Attendance and Grade Policies

8.1 Any late submissions for the assignment and coursework will NOT be accepted.

8.2 Over 15 minute late is class absence. Also, if students must be absent, please notify the lecturer before the date of their absence.

8.3 The student who does not take the final exam may fail this course.

8.4 The evaluation is based on a *curve* (the average score most students have received). The letter grades will be A, B+, B, C+, C, D+, D, or F.

8.5 The following letter grades may also be given:

"I" Incomplete

"P" In progress

"W" Withdrawn

9. Assignment and Examination Policies

Cheating in software engineering classes is a serious problem, as it devalues the hard work of honest students. Therefore, the department is determined to eliminate it by making th chances of getting caught high enough, and the penalties severe enough, that any student considering cheating will choose instead to just do their best, honestly, even if it means a low grade

Policy for cheating in SE courses

1. First offense for cheating on

- an exam (or quiz): zero on the exam (or quiz)
- a programming assignment or written homework: zero on the assignment and 10 points subtracted from final course total.

2. Second offense for cheating of any kind: failure in the course