

## แผนการสอนและการประเมินผล

รหัสวิชา	2301263	
ชื่อวิชา	โครงสร้างข้อมูล (Data Structures)	
หน่วยกิต	4 หน่วยกิต (3 หน่วยกิตบรรยาย และ 1 หน่วยกิตปฏิบัติการ)	
เงื่อนไขรายวิชา	prerequisite 2301262 co-requisite 2301233	
ชื่ออาจารย์ผู้สอน	บรรยาย: ผู้ช่วยศาสตราจารย์ ดร. พีระพนธ์ โสพัศสถิตย์ ปฏิบัติการ: อาจารย์ ดร. รชลิดา ลิปิกรณ	
เนื้อหารายวิชา	ชนิดข้อมูลพื้นฐาน ต้นไม้ ชนิดข้อมูลนามธรรม การเรียงลำดับ และการค้นหา และจัดการหน่วยความจำ Basic data types, trees, abstract data types, sorting and searching, and memory management.	
ประมวลลักษณะวิชา	เน้นทฤษฎีพื้นฐานการจัดเรียงข้อมูล เพื่อสะดวกในการจัดการ สืบค้น เก็บรวบรวมข้อมูล ในลักษณะต่างๆ โดยให้ตัวอย่างที่เข้าใจได้ง่ายสำหรับผู้เรียน นอกจากนี้ ยังมีการปฏิบัติทำตัวอย่างจริง ซึ่งต้องประยุกต์ทฤษฎีเหล่านั้นเข้าด้วยกัน เพื่อให้ผู้เรียนได้ทั้งทฤษฎี และทักษะที่สามารถนำไปประยุกต์ใช้ในหลักสูตรขั้นสูงต่อไป	
วัตถุประสงค์เชิงพฤติกรรม	นิสิตสามารถ	<ol style="list-style-type: none"> <li>1. แปลงโครงสร้างข้อมูล โดยใช้โครงสร้างเฉพาะแบบ ได้แก่ linked list, stacks, queues, trees, ADT, heaps, hashing techniques ได้</li> <li>2. ใช้ stack เพื่อเก็บค่าของ address และใช้คำนวณ polish notation ทำ parse tree</li> <li>3. ใช้ queues เพื่อจัดลำดับการเข้าและออกของข้อมูล</li> <li>4. ใช้ hashing technique ทำ table lookup และ searching ได้</li> <li>5. เขียนโปรแกรมแก้ปัญหาเกี่ยวกับ sets ได้</li> <li>6. เขียนโปรแกรมจัดลำดับข้อมูล โดยใช้เทคนิคต่างๆ เช่น quicksort, heapsort, mergesort</li> <li>7. เขียนโปรแกรมทำหน้าที่จัดการพื้นที่ในหน่วยความจำและเก็บข้อมูลที่ไม่ต้องการใช้แล้ว</li> </ol>
วิธีการสอน	<ul style="list-style-type: none"> <li>● บรรยาย อภิปรายโดยผู้สอน</li> <li>● ค้นคว้าหาข้อมูลจากหนังสืออ่านประกอบ</li> <li>● ประสพการณ์คอมพิวเตอร์ โดยการทำแบบฝึกหัด และโครงงานในภาคปฏิบัติ</li> </ul>	
การวัดผลการเรียน	สอบย่อย	10%
	สอบกลางภาค	25%
	สอบปลายภาค	35%
	แบบฝึกหัด/โปรแกรม	30%
	หมายเหตุ:	นิสิตที่ส่งแบบฝึกหัด/โปรแกรมน้อยกว่า 50% ถือว่าไม่ผ่าน ไม่ว่าจะทำคะแนนหมวดอื่นๆ ได้มากเท่าใด
เกณฑ์การตัดเกรด	86 - 100	A
	80 - 85	B+
	75 - 79	B
	70 - 74	C+
	60 - 69	C
	55 - 59	D+
	50 - 54	D
	0 - 49	F

บรรยาย	เนื้อหาและกิจกรรม
ชั่วโมงที่	
1 - 3	Introduction <ul style="list-style-type: none"><li>● Algorithm analysis and notation</li></ul>
4 - 6	Basic data types <ul style="list-style-type: none"><li>● Arrays, linked lists</li></ul>
7 - 10	Basic data types (cont'd) <ul style="list-style-type: none"><li>● Stacks, queues</li></ul>
11 - 13	Trees <ul style="list-style-type: none"><li>● Binary trees, AVL trees</li><li>● B-trees</li></ul>
14 - 17	Trees (cont'd) <ul style="list-style-type: none"><li>● Tree traversal</li><li>● heaps</li></ul>
18 - 20	Abstract Data Types <ul style="list-style-type: none"><li>● enqueue and dequeue; push and pop</li><li>● monitors</li></ul>
21 - 23	Graphs <ul style="list-style-type: none"><li>● topological sort</li></ul>
24 - 27	Graphs (cont'd) <ul style="list-style-type: none"><li>● network flow problems</li><li>● depth-first search</li></ul>
28 - 30	Hashing <ul style="list-style-type: none"><li>● separate chaining, open addressing</li><li>● rehashing</li></ul>
31 - 33	Sorting and searching <ul style="list-style-type: none"><li>● Internal sorting models: Insertion and selection sorts</li><li>● Quicksort</li></ul>
34 - 36	Sorting and searching (cont'd) <ul style="list-style-type: none"><li>● Heapsort</li><li>● Shell sort, bucket sort</li></ul>
37 - 40	Sorting and searching (cont'd) <ul style="list-style-type: none"><li>● Mergesort</li><li>● Linear and binary search</li></ul>
41 - 45	Memory management <ul style="list-style-type: none"><li>● Garbage collection algorithms</li><li>● Storage compaction, the Buddy systems</li></ul>

## ปฏิบัติการ

ครั้งที่ เนื้อหาและกิจกรรม

1. วิธีสร้าง data structures แบบ linked lists, stacks, queues พร้อมทั้งเขียนโปรแกรมเพื่อเข้าถึงข้อมูลใน data structures เหล่านี้
2. วิธีสร้าง data structure แบบ tree พร้อมทั้งเขียนโปรแกรมเพื่อเข้าถึงข้อมูลใน tree แบบ pre-order, post-order, in-order และ tree balancing กับ AVL tree
3. เขียนโปรแกรมประยุกต์การทำงานของ monitor ด้วยการเข้าและออกจากแถวคอยตามวิธีของ monitor
4. เขียนโปรแกรมประยุกต์การทำงานของ graph เช่นการทำ topological sort และ network flow ฯลฯ
5. เขียนโปรแกรมทำ hash table เพื่อการค้นหาที่รวดเร็ว
6. เขียนโปรแกรม sort แบบพื้นฐานโดยขั้นตอนวิธีต่างๆ
7. เขียนโปรแกรมทำ quicksort และ heapsort สำหรับข้อมูลซับซ้อนที่ไม่ได้เก็บในรูปของ array
8. เขียนโปรแกรมทำ mergesort, bucket sort
9. เขียนโปรแกรมทำการค้นหาแบบ linear และ/หรือ binary search และเปรียบเทียบกับแบบ hashing
10. เขียนโปรแกรมจัดการหน่วยความจำแบบพื้นฐาน โดยเลียนแบบ (simulate) การทำงานของระบบจัดสรรหน่วยความจำ (ด้วย array ขนาดใหญ่) ทั้ง allocation และ compaction

## รายชื่อหนังสือและเอกสารที่ใช้ประกอบ

1. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, Addison-Wesley, 1997.
2. Data Structures and Algorithm Analysis in Java, Mark Allen Weiss, Addison-Wesley, 1999.
3. Introduction to Algorithms, Cormen, T. H., Leiserson, C. E. and Rivest, R. L., MIT Press, 1990.
4. Data Structures & Their Algorithms, H. Lewis and L. Denenberg, Harper Collins Publishers, 1991.
5. Data Structures, Data Abstraction-A Contemporary Introduction Using C++, Mitchell L. Model, Prentice-Hall, 1994.
6. Computer Algorithms—Introduction to Design and Analysis, Sara Baase and Allen Van Gelder, Addison-Wesley, 2000.

สถานที่ติดต่อ

ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย โทรสาร: 02-255-2287

อ. พีระพนธ์ โสภัสสฤติย์ โทรศัพท์: 02-218-5158 email: Peraphon.S@chula.ac.th

อ. รัชลิดา ลิปิกรณ์ โทรศัพท์: 02-218-5150-1 email: rajalida.L@chula.ac.th

download เอกสาร

<http://pioneer.netserv.chula.ac.th/~sperapho/files/class/263.html>