

## CSCE 146: Algorithmic Design II

1. Course number and name: CSCE 146: Algorithmic Design II
2. Credit: 4-hrs; Contact: 2 lectures of 75 minutes each; 1 lab of 1 hour, 55 minutes each per week
3. Instructor:           Fall 2010: Jijun Tang  
                          Spring 2011: Duncan Buell
4. Text book:           Fall 2010: Simon Gray. *Data Structures in Java*, Pearson, 2007, ISBN: 9780321392794.  
                          Spring 2011: Duncan Buell. *Data Structures and Algorithms Using Java*, 2011, Jones and Bartlett, to appear (advance e-copy used).
5. Specific course information
  - a. Catalog description: Continuation of CSCE 145. Rigorous development of algorithms and computer programs; elementary data structures. Open to all majors.
  - b. Prerequisites: Grade of C or better in both CSCE 145 and MATH 141
  - c. Required in all curricula
6. Specific goals for the course
  - a. Specific outcomes of instruction are that students will be able to:
    1. Develop structured, modular algorithms.
    2. Implement correct programs in an object-oriented language.
    3. Use and implement as classes data structures, such as sets, bags, sequences, stacks, queues, and binary trees.
    4. Analyze the time and space complexity of simple algorithms.
    5. Apply data abstraction and elementary concepts of object-oriented programming.
    6. Implement moderately complex programs using an object-oriented language (presently Java).
  - b. Relation of course outcomes to Student Outcomes: CE: see page 2; CS & CIS: see page 3
7. Topics covered and approximate weight (14 weeks, 4 hours/week, 56 hours total)
  1. Overview of Object-Oriented Programming and Java (1 hour)
  2. Error Handling, Software Testing, and Program Efficiency (5 hour)
  3. Fundamental Data Structures: The Array and Linked Data Structures (5 hour)
  4. A Basic Collection Class (3 hour)

5. The List Abstract Data Type (5 hour)
6. The Stack Abstract Data Type (5 hour)
7. The Queue Abstract Data Type (5 hour)
8. Recursion (7 hour)
9. Sorting and Searching (7 hour)
10. Trees (7 hour)
11. Reviews and Exams (6 hour)
12. The Map ADT
13. Graphs

## Computer Engineering

### Relation of Course Outcomes to EAC Student Outcomes\*

Course Outcomes (CE)	Student Outcomes											
	(a) apply knowl edge of mathe matics , scienc e, and engine ering	(b) design and condu ct experi ments, ... interpr et data	(c) design a syste m, comp onent, or proces s to meet desire d needs ...	(d) functi on on multid iscipli nary teams	(e) identif y, formu late, and solve engine ering proble ms	(f) unders tandin g of profes sional and ethical respon sibilit y	(g) comm unicat e effecti vely	(h) broad educat ion to unders tand the impac t of engine ering soluti ons ...	(i) a recogn ition of the need for, and an ability to enga ge in life- long learni ng	(j) a knowl edge of contem porary issues	(k) use the techni ques, skills, and moder n engine ering tools ....	(CE) demo nstrate knowl edge of discret e mathe matics [CE]
Criteria	a	b	c	d	e	f	g	h	i	j	k	CE
1. Develop structured, modular algorithms.	1		3		3						1	
2. Implement correct programs in an object-oriented language.			3		3						1	
3. Use and implement as classes data structures, such as sets, bags, sequences, stacks, queues, and binary trees.	1	1	3		3						1	

4. Analyze the time and space complexity of simple algorithms.	3	2	2		3							3
5. Apply data abstraction and elementary concepts of object-oriented programming.			2		3							
6. Implement moderately complex programs written in an object-oriented language.			3		3						1	

\* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related

c.

## Computer Science & Computer Information Systems

### Relation of Course Outcomes to CAC Student Outcomes\*

<b>Course Outcomes (CS &amp; CIS)</b>	<b>Student Outcomes</b>											
	<b>All</b>									<b>CS</b>		<b>CIS</b>
	(a) apply knowledge of computing and mathematics appropriate to the discipline	(b) analyze a problem, and identify and define the computing requirements ...	(c) design, implement, and evaluate a computer-based system, ...	(d) function effectively on teams to accomplish a common goal	(e) An understanding of professional, ethical, legal, ... responsibilities	(f) communicate effectively with a range of audiences	(g) analyze the local and global impact of computing on ... society	(h) Recognition of the need for ... continuing professional development	(i) current techniques, skills, and tools necessary for computing practice	(j) apply mathematical foundations, algorithmic principles, and CS theory ...	(k) apply design and development principles	(l) An understanding of processes that support the information systems environment.
Criteria	a	b	c	d	e	f	g	h	i	j	k	l
1. Develop structured, modular algorithms.	1	1	3						1		1	
2. Implement correct programs in an object-oriented language.		2	3						1	1	3	
3. Use and implement as classes data structures, such as sets, bags, sequences, stacks, queues, and binary trees.	1	2	3						2	1	3	2
4. Analyze the time and space complexity of simple algorithms.	3	2	2						1	3	2	1
5. Apply data abstraction and elementary concepts of object-oriented programming.		1	2						2		2	1
6. Implement moderately complex programs written in an object-oriented language.		2	3						1	1	3	1

\* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related

