

COURSE DESCRIPTION

| | | | |
|--------------------|----------|---------------------|--------------------------|
| Course No. | CSC 330 | Course Title | Advanced Data Structures |
| Course Type | Required | | |
| Sem. Hours | 3 | Coordinator | Lixin Fu |

Current Catalog Description:

Static and dynamic data structures emphasizing binary trees and graphs. Advanced programming techniques. Advanced sorting and searching algorithms. Hashing techniques. Performance analysis. Methods of developing large applications programs.

Textbook:

Data Structures & Problem Solving Using Java, 4th Edition, Mark Allen Weiss, Addison-Wesley, ISBN-13: 978-0-321-54140-6 or ISBN-10: 0-321-54140-5.

References:

None

Course Outcomes:

Students should be able to:

1. understand and use advanced features of Java such as inheritance and generics
2. understand and use linear and non-linear data structures
3. have a solid introduction to advanced programming techniques and performance analysis
4. greatly improve their programming ability through the study of advanced data structures and through programming practice

Activities Enabling Program Outcomes (POx refers to program student outcome x)

Instruction: The core of this course involves exploration of Abstract Data Types and Data Structures. ADTs are introduced as abstractions specified using interfaces and object-oriented design (POk), and data structures are presented as solutions with operations that are mathematically analyzed for correctness and efficiency (POa and Poj). Discussion and tips about how data structure selecting fits into the overall software design and implementation process are included in class (POc, POi, and POk).

Student Activities and Assessment: Every offering of this course will include (details of assessment criteria and expectations are in outcome rubrics):

1. At least one written assignment or program in which students are required to select an appropriate data structure and evaluate the time complexity of their solution using appropriate mathematical techniques (POa and POj)
2. At least one programming assignment in which students start with a basic design document that identifies required tasks and computational requirements (POb)
3. At least one program in which students must use abstraction and object-oriented design and implement an effective solution (POi and POk)
4. One or more exam questions which require students to demonstrate use of C.S. and mathematical terminology, and perform mathematical analysis of operations on data structures (POa)

Prerequisites by Topic:

| |
|--|
| Students must have |
| <ul style="list-style-type: none"> • a grade of at least C (2.0) in CSC 230 (Elementary Data Structures and Algorithms) and • CSC 250 (Foundations of Computer Science I) |

Major Topics Covered in the Course:

| |
|--|
| <ul style="list-style-type: none"> • Java advanced features: inheritance, files, collections, functional object • Algorithms: recursion, divide-and-conquer, backtracking, dynamic programming, greedy, and applications to sorting, compression, etc. • Binary trees: concepts, representations and traversals, heaps, search trees, 2-3-4 trees, red-black trees • Associative structures: sets, maps, hash tables • Graphs: concepts, representations and traversals |
|--|

Estimated Curriculum Category Content (Semester hours):

| <i>Area</i> | <i>Core</i> | <i>Advanced</i> | <i>Area</i> | <i>Core</i> | <i>Advanced</i> |
|----------------------------|-------------|-----------------|------------------------|-------------|-----------------|
| Algorithms | 1 | 0 | Software design | 0 | 0 |
| Data structures | 2 | 0 | Prog. Languages | 0 | 0 |
| Comp Org & Arch | 0 | 0 | | | |