

Dr. Dichter
CS 329

University of Bridgeport

Office: Technology Building 223

Office Hours:

Phone: 576 - 4763

email: dichter@bridgeport.edu

website: www.bridgeport.edu/~dichter

Text: Kingston, *Algorithms and data Structures*, Addison Wesley

ISBN 0-201-40374-9

<http://www.aw.com/catalog/academic/product/1,4096,0201403749,00.html>

The course focuses on data algorithm analysis algorithm design and analysis. Many important algorithms will be studied and implemented. Complexity, performance and correctness considerations of algorithms will be studied. The student is expected to be proficient in programming in the C++ language and familiar with OO design and have mathematical level of proficiency including discrete mathematics and calculus I, II.

There will be several machine assignments, an examination and a final project. All assignments must be handed in at the beginning of the class on the due date to receive full credit. All assignments and exams must be completed to pass the course.

Determination of Grade:

Homework	-	10%
Midterm Exam	-	30%
Final Exam	-	30%
Programming	-	30%

note: All machine assignments must be handed in according to the following specifications:

- Modular coding is required, using appropriate OO concepts
- Include accompanying documentation with a clear program statement and explanation of each module. Use pre and post conditions for functions and methods.

Topics

- Mathematical methods for analysis: discrete and continuous measures of algorithmic complexity.
- Proofs of Correctness: inductive proofs, proofs by contradiction
- Time Complexity of algorithms, Big O, Big Θ
- Analyzing algorithm complexity and Amortized Complexity
- String Algorithms: KMP substring, and Non-redundant permutations
- Graphs: directed and undirected graphs. BFS, DFS, least-cost path algorithms
- Algorithm Design Techniques: Dynamic Programming, Recursive, Iterative
- Genetic Algorithms
- Sorting algorithms: Average complexity analysis
- Trees : general, binary, BST. Huffman Trees, B-Trees, Heap Trees
- Recursive Algorithms: Fast Fourier Transform in Polynomial Multiplication
- Encryption Algorithms: RSA
- Linear Programming Methods
- Introduction to Parallel Algorithms