CS 102 Introduction to Computing II

Department of Computer Science and Engineering

University of Bridgeport

Stamford Campus

Spring 1999

Catalog Data:

Introduction to data structures and algorithms. Top-down design and structured programming, debugging. Arrays and indexing, stacks, queues, lists, linked lists, trees, hash tables. Searching and sorting. Prerequisite: Computer Science 101. *3 lecture hours, 3 semester hours.*

Textbook:

Thomas A. Standish, *Data Structures, Algorithms & Software Principles in C.* Addison Wesley Publishing Company.

References:

Mark A. Weiss, *Algorithms, Data Structures, and Problem Solving with C++*, Addison-Wesley Publishing Company, 1996.

Jean-Paul Tremblay and Paul G. Sorenson, *An Introduction to Data Structures with Applications.* Second Edition, McGraw-Hill Publishing Company.

Thomas A. Standish, *Data Structures, Algorithms & Software Principles in C.* Addison Wesley Publishing Company, 1995.

Christopher J. Van Wyk, *Data Structures and C Programs.* Addison-Wesley Publishing Company, 1992.

Aaron M. Tanenbaum, *Data Structures using C.* Prentics Hall, 1990.

Jeffrey Esakov, *Data Structures: An advanced approach using C.* Prentice Hall, 1989.

Brian W. Kernighan and Dennis M. Ritchie, *The C Programming Language.* Prentice Hall.

S. E. Goodman and S. T. Hedetniemi, *An Introduction to the Design and Analysis of Algorithms.* McGraw Hill, 1985

A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data Structures and Algorithms.* Addison-Wesley, 1983.

E. Horowith, and S. Sahni, *Fundamentals of Data Structures.* Computer Science Press, 1976.

Coordinator:

Tarek M. Sobh, Ph.D., P.E., CMfgE
Associate Professor and Director of External Engineering Programs

Room: 256 Dana Hall, Tel: 576-4116

e-mail: sobh@cse.bridgeport.edu

Web: http://www.bridgeport.edu/∼sobh

Web: http://www.bridgeport.edu/∼risc

Goals:

This course is an introduction to data structures and algorithms. The emphasis is on on understanding and implementing major CS algorithms and applications. There will be a number of student demonstration sessions throughout the class. Algorithm design and complexity analysis will also be studied.

Prerequisites by Topic :

Ability to program in C. Understanding of structured design, data types, subprograms; arrays and records, top-down programming, algorithm development, pointers, basic recursion, and parameter passing.

Meeting Times:

Tuesday 6:00 - 9:00 (except for 5/25/99 to be moved to Thursday 5/20/98, and 5/11/99 is cancelled)

3/16, 3/23, 3/30, 4/6, 4/13, 4/20, 4/27, 5/4, 5/18, 5/20

Topics and Dates:

- Overview of Data Structures and Algorithms.
- Introduction to Complexity Analysis, progressions and recurrence equations, efficiency of algorithms and data structures.
- Storage Structures and Indexing for Arrays.
- Linked Data Representations, Linked Linear Lists and other Linked Data Structures.
- Introduction to Recursion.
- Modularity, Priority Queues, and Data Representation.
- Top-Down Programming, Program Correctness, Optimizing Programs, Complexity of Algorithms, O-Notation.
- Linked Data Structures – Stacks and Queues ADT.
- Trees ADT, BST, Heaps and Priority Queues.
- Graphs, Directed and Undirected graphs, BFS, DFS, Graph searching, Shortest Paths.
- Sorting and Searching Algorithms, Design and Analysis of different methods.
- Hashing, Table ADT, Advanced Recursion Techniques and Parsing.

Policies and Grading:

Grade Calculation:

- Quizzes and Participation: 50%
- Homeworks & Laboratory Projects (Machine Assignments): 50%

Policies:

- Late homeworks and programs *will not be accepted*.
- Three or more unexcused absences will result in an automatic failure.
- Make-up exams will not be allowed (except for prior instructor approval for a documented emergency)
- Homeworks and programs are due within a week from the assignment date, unless the instructor notes otherwise.
- All homeworks are to be *typed*.
- Structured and modular code is required for the machine assignments.
- All programs are to include sufficient comments and documentation with a clear program statement.
- Programs are to be submitted with all I/O files and source code hardcopies, in addition to a disk version, a demonstration is required for all programming homeworks.
- Extra credit quizzes, assignments, and programs (if any) will be announced by the instructor.
- The instructor will discard the worst (lowest score) one or two quizzes for each student at the end of the class (they will not be counted towards calculation of the final grade).