Course Assessment Document for CS 231 Data Structures and Algorithms

Departmental Outcomes

1. Proficiency in computational thinking
2. Ability to analyze systems at the three levels of computer science: theory, software, and hardware
3. Proficiency in the design and implementation of algorithms using multiple programming languages
4. Ability to apply computational thinking to a diverse set of problems and disciplines
5. Ability to communicate effectively and collaborate with others
6. Ability to adapt to new challenges and computational environments

Course Description

The course focuses on the common structures used to store data and the standard algorithms for manipulating them. Standard data structures include lists, stacks, queues, trees, heaps, hash tables, and graphs. Standard algorithms include searching, sorting, and traversals. Along with implementation details, students will learn to analyze the time and space efficiency of algorithms and how to select appropriate data structures and algorithms for a specific application. In homeworks, labs and programming projects, students will implement their own data structures and make use of existing libraries to solve a variety of computational problems.

Prerequisites: a grade of C- or better in CS 151 or permission of instructor.

Rationale for prerequisites: The course builds upon the computational thinking and programming skills students learn in CS 151. Students with prior experience with a programming language may skip CS 151 and begin with this course.

Desired Course Outcomes

A. Students understand the advantages and disadvantages of fundamental data structures and can implement them using object-oriented design principles.

B. Students understand, can implement, and can calculate the time and space efficiency of classic search, sort, and traversal algorithms, including the use of big-Oh notation.

C. Students understand the tradeoffs between different implementation of data structures and algorithms and can make appropriate design decisions based on application data requirements.

D. Students can use fundamental data structures and algorithms appropriately to solve a variety of computational problems.

E. Students can communicate the result of their work and describe an algorithm.

We will disseminate the desired course outcomes to students via the course web page, syllabus and in class.

Course Matrix

Outcome	Activities	Method of Assessment	Departmental Outcome
A	Lectures, Homeworks, Labs, Projects	Exams, graded assignments	1, 2, 3
B	Lectures, Homeworks, Labs, Projects	Exams, graded assignments	1, 2, 3
C	Lectures, Homeworks, Labs, Projects	Exams, graded assignments	1, 2, 3
D	Homeworks, Labs, Projects	Exams, graded assignments	1, 3, 4, 6
E	Project reports	Graded projects, class participation	5

Grade Calibration Matrix

Outcome	Meaning of the grade A
A	The student knows each data structures' uses and can implement the data structures using array-based and reference-based approaches as appropriate.
B	The student knows basic sorting, searching, and traversal algorithms and can implement them using iteration and recursion. The student can calculate the algorithms' time and space complexities.
C	The student can weigh the advantages and disadvantages of different implementations of the data structures and algorithms discussed in class, and can use these factors in making program design decisions.
D	The student can analyze a problem, determine the most appropriate data structure to solve the problem, and implement an elegant solution.
E	Reports are well-written, concise, and clear. The reports clearly describe the project and use example code to demonstrate concepts and design decisions. The reports are written so that students outside the course could understand them.

Outcome	Meaning of the grade B
A	The student knows each data structures' uses and can implement most of them using array-based and reference-based approaches as appropriate.
B	The student knows most of the basic sorting, searching, and traversal algorithms and can implement them using iteration or recursion. The student knows the algorithms' time and space complexities.
C	The student understands the advantages and disadvantages of different implementations of the data structures and algorithms discussed in class, and can weigh these factors in making some program design decisions.
D	The student can analyze a problem, determine an appropriate data structure to solve the problem, and implement a solution.
E	Reports are well written and clear. The reports describe the main points of the project, mention design issues and may make use of example code to support the text. The reports are written so that students taking the course could understand the work.

Outcome	Meaning of the grade C
A	The student understands each data structure and can implement some of them using array-based and reference-based approaches as appropriate.
B	The student knows most of the basic sorting, searching, and traversal algorithms and can implement some of them using iteration or recursion. The student knows most of the algorithms' time and space complexities.
C	The student understands the advantages and disadvantages of different implementations of the data structures and algorithms as discussed in class, but may have difficulty selecting appropriate data structures.
D	The student can analyze a problem, determine an appropriate data structure to solve the problem, but has trouble implementing a solution.
E	Reports describe the work, but may take the form of a narrative of the student's difficulties rather than focusing on the results. The reports assume the reader has in-depth knowledge of the project.

Outcome	Meaning of the grade D
A	The student has basic familiarity with each data structures and can implement some of them.
B	The student has familiarity with most of the basic sorting, searching, traversal algorithms, can understand how they work by looking at their code, and can implement some of them. The student knows some of the algorithms' time and space complexities.
C	The student understands the advantages and disadvantages of different implementations for some of the data structures and algorithms as discussed in class, but has difficulty making design decisions.
D	The student has difficulty analyzing a problem and determining which data structure is most appropriate.
E	The student's reports are incomplete or not well written and may contain errors. The report includes some description of the student's work, but little explanation.

A student who receives an F does not meet the criteria for a D or any higher grade.