

NOVA COLLEGE-WIDE COURSE CONTENT SUMMARY  
CSC 202 – COMPUTER SCIENCE II (4 CR.)

**Course Description**

Examines fundamental data structures and analyzes algorithms. Covers abstract data types and essential data structures such as arrays, stacks, queues, linked lists, and trees; introduces searching and sorting algorithms and algorithm analysis. Lecture 4 hours per week.

**General Course Purpose**

Examines data structures and analyzes algorithms. Covers data structures (including sets, strings, stacks, queues, arrays, records, files, linked lists, and trees), abstract data types, algorithm analysis (including searching and sorting methods), and objects.

**Course Prerequisites/Corequisites**

Prerequisite: Prerequisite: CSC 201 and MTH 263.

Corequisite: MTH 264 or division approval.

**Course Objectives**

Upon completion of this course, the student will be able to:

- A. Analyze Problems
- B. Develop Algorithms in an Object Oriented Environment

**Major Topics to be Included**

- I. Data abstraction and object-oriented programming (review)
  - A. dynamic allocation
  - B. overloading functions and operators
  - C. constructors and destructors
- II. Recursion
  - A. analysis
  - B. applications
- III. ADT implementations (static and dynamic) and applications
  - A. linked lists (singly, doubly, circular)
  - B. stacks
  - C. queues
  - D. trees and tree traversal
- IV. Searching
  - A. analyzing and comparing searching methods
  - B. relationships between methods and data structures
- V. Sorting
  - A. examples of order  $n^2$  and order  $n \log(n)$  sorts
  - B. analyzing and comparing sorting methods

VI. Optional other topics

- A. hashing
- B. tables
- C. file structures
- D. sets
- E. Templates

**5. Suggested Time Allocation per Topic**

In order to standardize the core topics of CSC 202 so that a course taught at one campus is equivalent to the same course taught at another campus, the following student contact hours per topic are recommended. Of course, the topics cannot be followed sequentially. Many topics are taught best as an integrated whole, often revisiting the topic several times, each time at a higher level. There are normally 64 student-contact-hours per semester for a four-unit course. The last category, *Other/Enhance*, leaves ample time for an instructor to tailor the course to special needs or resources.

<b>Ref</b>	<b>Topic</b>	<b>Hours</b>	<b>Percent</b>
I	Data abstraction and object-oriented programming	10	16
II	Concept of Recursion	2	3
III A	Linked lists/pointers	10	16
III B	Stacks	8	13
III C	Queues	8	13
III D	Trees	2	3
IV	Searching	4	7
V	Sorting	6	9
VI	<i>Other</i> optional content or <i>Enhance</i> the above	8	13
	Exams and quizzes	6	9
	Total	64	100

l.