

**Endicott College**  
**Beverly, Massachusetts**  
**School of Arts and Sciences**  
**Computer Science Department**  
**Course Syllabus**

<b>Course #:</b>	CSC 161.01
<b>Course title:</b>	Data structures and algorithms
<b>Credits:</b>	3
<b>Pre-requisites:</b>	CSC 160 or equivalent; C++ proficiency
<b>Semester:</b>	Spring 2014
<b>Meeting times:</b>	Lectures M/W 2–3:15pm in LSB 312
<b>Textbook:</b>	None required; Recommended: <ul style="list-style-type: none"><li>• <i>Algorithms in C++</i> (Sedgewick, 1993)</li><li>• <i>C++ How to Program</i> (Deitel &amp; Deitel, 2013)</li><li>• <i>Programming Principles and Practice using C++</i> (Stroustrup, 2009)</li></ul>
<b>Web page:</b>	<a href="http://hank.feild.net/courses/2014-sp/csc161">http://hank.feild.net/courses/2014-sp/csc161</a>
<b>Instructor:</b>	Henry Feild, Ph.D.
<b>Office:</b>	LSB 113A
<b>Email:</b>	<a href="mailto:hfeild@endicott.edu">hfeild@endicott.edu</a>
<b>Phone:</b>	x7793
<b>Office hours:</b>	T/Th 2–3:30pm and by appointment

## Catalog Description

This course focuses on expanding the fundamental programming concepts learned in CSC 160. Students will learn about elementary abstract data types including classes, linked lists, stacks, queues, heaps, trees and pointers. We will also analyze fundamental computing algorithms, including searching and sorting, as well as recursion. Laboratory exercises in C++. *Required for CSC majors.*

## Learning Outcomes

At the completion of this course the student should be able to:

- apply a divide and conquer approach to leverage recursion for to solve selected computational problems
- select appropriate data structures to aid in efficiently solving computational problems
- evaluate the computational complexity of data structures and algorithms

## Teaching/Learning Strategies

This course will consist of a mixture of lectures, labs, and discussions. Students are expected to complete homework, programming assignments, and in-class exercises.

## Outline

Week	Dates	Topic
1	Jan. 27, 29	C++ review; OOP (inheritance, polymorphism); Exceptions
2	Feb. 3, 5	Recursion
3	Feb. 10, 12	<b>Quiz 1</b> ; Searching; Complexity analysis
	<i>Feb. 17</i>	<i>No class—Presidents day</i>
4	Feb. 19	Sorting
5	Feb. 24	More sorting
	<b>Feb. 26</b>	<b>Exam 1</b>
6	March 3, 5	ADTs: Arrays, Lists, and Vectors
7	March 10, 12	<b>Quiz 2</b> ; Stacks and Queues
	<i>March 18, 20</i>	<i>No class—Spring break</i>
8	March 24, 26	Trees
9	March 31, April 2	More trees
10	April 7	Priority queues
	<b>April 9</b>	<b>Exam 2</b>
11	April 14, 16	Hash tables
	<i>April 21</i>	<i>No class—Patriots day</i>
12	April 23	Tries
13	April 28, 30	<b>Quiz 3</b> ; Graphs
14	May 5, 7	More graphs
15	May 12	Review
	<b>FEP</b>	<b>Final Exam</b> (cumulative)

## Grading

<b>Exam 1</b>	15%	<b>Quizzes</b>	20%
<b>Exam 2</b>	15%	<b>Programming Assignments</b>	25%
<b>Final Exam</b>	20%	<b>Participation</b>	5%

100–94 = <b>A</b>	89–87 = <b>B+</b>	79–77 = <b>C+</b>	69–67 = <b>D+</b>	59–0 = <b>F</b>
93–90 = <b>A-</b>	86–83 = <b>B</b>	76–73 = <b>C</b>	66–63 = <b>D</b>	
	82–80 = <b>B-</b>	72–70 = <b>C-</b>	62–60 = <b>D-</b>	

## Required Texts/Technology

There are no required texts for this class. There are a couple of suggested texts (see above). These texts may be helpful for students to see the material presented in a different way from

how it is covered in class. Technology is not required as LSB 312 is equipped with the necessary software. However, students are encouraged to set up their computers with a text editor and C++ compiler in order to work on assignments outside of LSB 312.

## **Late Policy**

Each student will be given four (4) floating late days for programming assignments and labs throughout the semester. Using a late day will not result in any late penalization. Each late day is a 24 hour period—if an assignment is due Tuesday at 5pm, using one late day makes the assignment due Wednesday at 5pm; using two makes the due date Thursday at 5pm, etc. Homework submitted late, or assignments/labs submitted late after all late days have been used, may not be collected and a grade of 0 may be automatically assigned. It is the student's responsibility to keep track of used late days.

## **ADA Policy**

If you as a student qualify as a person with a disability as defined in Chapter 504 of the Rehabilitation Act of 1973, the Americans with Disabilities Act (ADA) of 1990, the Americans with Disabilities Act Amendments Act of 2008 (ADAAA), you are strongly encouraged to register with the Center for Teaching and Learning. The Center for Teaching and Learning is located in the Diane M. Halle Library room 201 and online at <http://www.endicott.edu/academicresources>. As a student registered with the Center for Teaching and Learning, it is your responsibility to present your accommodation letter to your instructor at the beginning of each semester.

## **Course Expectations**

For each credit hour, students are expected to spend a minimum of two hours on work out-side of class each week. For this three credit course, that is a minimum of six hours each week.

Students must review the Academic Calendar published by the Registrar's Office online at: <http://www.endicott.edu/Academics/AcademicCalendar.aspx>. Class attendance is expected of all students up to and including the last day of scheduled classes in the semester. Students must plan accordingly.

## **Attendance/Participation**

Students should attend every class. However, students may miss up to three classes excused or unexcused, without notifying the professor. In-class exercises cannot be made up for credit and office hours will not be used to cover material missed due to an absence. Participation will also be affected. **Absences beyond three classes may result in up to a 5% penalty on the final course grade per absence.** Please contact the professor in the event of extenuating circumstances resulting in the need for a prolonged absence.

Participation, which includes in-class exercises, is worth 5% of the final grade. Students who miss class, do not participate when in class, and consistently perform poorly on in-class

exercises will receive an overall participation grade of 0. Students who always come to class, are eager to participate, and consistently perform well on in-class exercises will receive the full 5 participation points, **and may earn up to 5 points extra credit.**

While in class, students are expected to be fully present and engaged. Using phones or lab computers in class for any non-class purpose—e.g., texting, making calls, checking email, watching videos, etc.—is strictly forbidden. Laptops are not to be used in class unless otherwise indicated. Side discussions, covert texting, or any other failure to pay attention will negatively impact your grade. Violators will be warned once, and asked to leave the lecture thereafter. Being asked to leave will be counted as an absence, regardless of when during the class the incident occurs. Consistent violations will result in failure of/dismissal from the class.

## **Working with Others**

You may discuss assignments with other students, but you *may not* share code, or view another student's assignment code prior to submission. *All code you submit must be your own.* If you are having trouble with a concept and would like help from a classmate, talk it out, draw it out, or come up with a similar example to code up that is not part of the assignment itself. Submitting someone else's code, in part or in whole, will be considered a violation of the Academic Integrity Policy (see below). Keep in mind, it is generally quite easy to detect when the same code has been submitted by two students, even when the time is spent to make the code look different. *Also note, you must be able to explain all of your submitted code to the professor if asked.*

## **Academic Integrity**

Students are expected to abide by the *Academic Integrity Policy* of Endicott College. Cheating will result in failure of the assignment or course or dismissal from the College. Make sure to always cite sources and if you confer with classmates on an assignment, list who those individuals are at the top of your submission. You are expected to be capable of explaining any code you submit to the professor when asked. Submitting identical or near identical assignments, submitting code that is not your own (whether or not you indicate whose it is), not making clear who you consulted with, or not being able to explain your submitted code when prompted will all be considered a violation of the Academic Integrity Policy.

## **Subject to Change**

This syllabus is subject to change at the discretion of the professor. Updates will be announced and the most recent version will be available on the course website.