

## ***Syllabus***

### **CST 238 – Introduction to Data Structures**

**California State University, Monterey Bay**

**Fall 2013**

#### **Course Information**

Credits: 4.0

Prerequisites: CST 231 or Instructor's consent

Lecture: TR, 10:00A – 10:50A, Building 18, Room 104

Lab: TR, 11:00A – 11:50A, Building 18, Room 104

#### **Course Description**

This is an intermediate-level programming course with emphasis on the data structures. Coverage includes the C++ basics, pointer, dynamic memory, class, I/O stream, list, stack, queue, recursion, tree, sorting, searching, and algorithm analysis.

#### **Instructor**

Name: YoungJoon Byun, Ph.D.

Email: ybyun@csumb.edu

Phone: 831-582-3602

Office: Bldg. 18/Room 134

#### **Office Hours**

Monday 2:00P – 3:00P

Tuesday 1:00P – 2:00P

Wednesday 2:00P – 3:00P

Or by appointment

#### **Course Web Site**

<http://ilearn.csumb.edu/>

Additional course information and announcements will be available on this site. It is student's responsibility to check this site frequently.

#### **Required Text**

Author: Larry Nyhoff

Title: ADTs, Data Structures, and Problem Solving with C++, 2<sup>nd</sup> Edition

Publisher: Prentice-Hall, 2005

ISBN: 0-13-140909-3

#### **Course Objectives**

The main objective of this course is to provide students with intermediate-level programming skills to make it possible for them to write clear, efficient, and high-quality code. To meet this objective, students will learn basic data structures such as array, list, stack, queue,

and tree. They will also acquire several basic algorithms in programming such as searching, sorting, and recursion. Moreover, students will have in-depth knowledge of C++ programming language such as classes, pointers, and dynamic memory management.

### **Course Outcomes**

This course is designed to enable students to complete the ITCD T2 tech specialization outcomes of software design issues. The T2 outcomes are “Analyze a problem and select the most appropriate data structure to represent the solution”, “Construct and implement an algorithm or set of algorithms which solve a problem and analyze the efficiency of those algorithms”, and “Describe the different system life cycle models and, given an application, explain which model would be appropriate for that application”.

In particular, upon successful completion of this course, student should be able to

- Define and use pointers in C++ program.
- Process data using dynamic memory allocation.
- Demonstrate the usage of I/O streams and strings in C++ program.
- Define and use programmer-defined classes.
- Demonstrate the usage of fundamental data structures such as list, stack, queue, and tree in C++ programming.
- Solve a programming problem using the recursion technique.
- Measure the efficiency of an algorithm.
- Use well-developed algorithms in sorting and searching.
- Solve a simple real world problem in C++ programming.

### **Main Topics**

- C++ review
- Pointers
- Dynamic memory
- I/O streams and strings
- Abstract data types, classes, and objects
- List, stack, queue, and tree
- Algorithm analysis and efficiency
- Sorting and searching
- Recursion
- Advanced topics, if time permitted

### **Exams**

- First midterm: October 3, 2013 (Thursday) in class.
- Second midterm: November 14, 2013 (Thursday) in class.
- Final: December 17, 2013 (Tuesday) in class.
- All exams are closed book.
- Final exam will be comprehensive.
- A calculator is not necessary for the exam.

- No re-grading will be accepted one week after the graded assignment is returned to the students.
- No makeup exam will be allowed, except in extreme emergency cases. Students are advised to let the instructor know beforehand, if possible.

### **Quizzes**

- Quizzes will irregularly be held in class.
- Quiz date will be announced in advance a few days before the quiz.
- No re-grading will be accepted one week after the graded assignment is returned to the students.
- No makeup for a quiz.

### **Homework and Project**

- Programming projects and problem solving assignments will be given throughout the semester.
- Late submission will be penalized 15% after the due date. Assignments handed in more than 48 hours later will not be accepted.
- No re-grading will be accepted one week after the graded assignment is returned to the students.

### **Labs**

- Microsoft Visual Studio will be used in the labs.
- A handout of each lab will be given before starting the lab.

### **Grading Policy**

Midterm – I	15%
Midterm – II	15%
Final Exam	20%
Project	15%
Homework	20%
Quiz	5%
Lab	10%

Grade		
A	100	93 ≥
A–	93 <	90 ≥
B+	90 <	87 ≥
B	87 <	83 ≥
B–	83 <	80 ≥
C+	80 <	76 ≥
C	76 <	70 ≥
F	70 <	0

### **Attendance**

Students are expected to be regular and punctual in class attendance. Students are responsible for all materials covered in lectures and labs. Lab attendance is mandatory. So, if you miss a lab, you will get zero point for the lab.

### **Note to Students with Disabilities**

Students with disabilities who may need accommodations please see the instructor as soon as possible during office hours or make an appointment by calling 582-3602 or by email [ybyun@csumb.edu](mailto:ybyun@csumb.edu). Also, contact:

Student\_Disability\_Resources@csumb.edu  
Building 47, Student Services, First Floor  
Phone: 831/582-3672 voice, or 582-4024 fax/TTY  
<http://sdr.csumb.edu/>

### **Academic Integrity Policy in This Class**

Cheating of any kind will not be tolerated at all in this class. You are responsible for your own original work on all assignments, labs, homework, exams, and all programming code. Two or more students submitting extremely similar code will receive a zero on the assignment and be reported to the academic dishonesty board. There are continual checks of the assignments to be sure that everyone is handing in original code.

For more detailed information, refer to <http://csumb.edu/site/x2161.xml>