CS 132 – Computer Science II
Course Syllabus

Spring, 2014
Dr. Steven K. Andrianoff

Time and Place

Lecture:	MWF	10:30 – 11:20	Walsh 204
Lab:	Tu	2:30 – 4:30	Walsh 101

Instructor

Dr. Andrianoff (andrianoff@sbu.edu)
Office: Walsh 113
Phone: 375-2053
Office hours:

Mon	1:30 – 2:30 p.m.
Tues	1:00 – 2:00 p.m.
Wed	11:30 – 12:30 p.m.
Thur	1:00 – 2:00 p.m.
Fri	1:30 – 3:00 p.m.

(Other times by appointment)

Course web page: http://www.cs.sbu.edu/andrianoff/CS132/index.htm

In general, announcements, readings, assignments, and laboratory exercises for CS 132 will be given in class AND published on the course web page. Students are expected to check that page regularly for news, and are nonetheless responsible for any assignment announced in either manner.

Text: Goodrich, Michael T., and Roberto Tamassia, *Data Structures & Algorithms*, 5th ed., John Wiley & Sons, Inc., (2010). ISBN 978-0-470-38326-1

Catalog Description:

This is the second course in the computer science major sequence. The course utilizes the object-oriented design approach to building applications, which emphasizes the creation and utilization of reusable software tools. Students are introduced to data structures that are commonly encountered in building software applications and to the analysis of the efficiency of algorithms used to solved problems. The programming language Java is used to implement software designs. The course consists of three lecture hours and one two-hour laboratory per week. 4 credits.

Prerequisites:

CS 131 – Computer Science I

Note: The course may not be taken for credit without the laboratory component.

Course Objectives

1. To consolidate the knowledge of algorithm design and programming that was gained in the first semester course, CS 131. The emphasis will be on the design and implementations of larger programs than encountered in CS 131.
2. To begin a detailed study of data structures and data abstraction as exemplified by classes and objects in Java. The collection classes available in the Java library (*Java API*) will be included in this study.
3. To introduce the uses of mathematical tools in algorithm analysis (O-notation).

Tentative schedule

Object-Oriented Design Principles (Chapter 2)	2 weeks
Inheritance and Class Hierarchies (Chapter 2)	2 weeks
Collections/Lists (Chapters 3 & 6)	2 weeks
Algorithm Efficiency and Big-O (Chapter 4)	1 week
Stacks/Queues (Chapter 5)	1 week
Searching/Sorting (Chapter 11)	2 weeks
Collections/Sets and Maps (Chapter 9)	2 weeks
Trees (Chapters 7 & 10)	2 weeks

Course Policies:

Course Grade

The final grade will be determined by three components: quizzes, final examination, and assignments (including laboratory assignments, programming assignments, and written homework assignments). The three components are weighted as follows:

Quizzes	30%
Final examination	30%
Assignments	40%

Note: A passing grade (minimum of 60%) must be earned from the quizzes and final examination for a student to receive a passing grade for the course.

Quizzes

Quizzes will be twenty-minute, closed-book exams held during class. Approximately eight quizzes will be given, however, only the five best quiz grades will contribute to the final grade, the others will be dropped. There will be no make-ups for unexcused absences when quizzes are given.

Final Examination

The final examination will be a 2-hour closed book examination. It is scheduled for Tuesday, May 6, at 8:00 a.m. Every student is required to take the final examination. The final examination is comprehensive.

Laboratories and Programming Assignments

There is a two-hour lab each week. Most labs require a write-up that must be done using a word processor and have a cover sheet. The penalty for a late lab write-up is 10% per day late. Lab write-ups will be due Monday at class time following the week assigned. **Attendance at laboratory sessions is mandatory: each absence over 1 will result in a reduction in grade by one letter grade modifier for the course**. Any material from a missed lab must be handed in on the date due and is subject to the same late penalty.

Programming assignments are assignments that are completed individually by each student outside of formal lab. The late penalty for programming assignments is the same as for labs – 10% per day. Please note that both labs and programming assignments are governed by the department’s policies and procedures distributed with this syllabus.

Written Homework Assignments

Homework assignments are due at the beginning of the class period on the day due. The penalty for late homework assignments is 10% per day until the time the graded work is returned by the instructor. No late assignments are accepted after the graded work is returned.

Attendance

There is no attendance requirement for lectures, however attendance will be monitored. Students are expected to attend all of the classes and will be responsible for all assignments. More than three absences is considered excessive.

Attendance at laboratory sessions is mandatory. **You are allowed one absence from lab. Each absence over 1 will result in a reduction in grade by one letter grade modifier for the course.**

Academic integrity policy

Academic dishonesty is inconsistent with the moral character expected of students in a university committed to the spiritual and intellectual growth of the whole person. It also subverts the academic process by distorting all measurements. It is a serious matter and will be dealt with accordingly. A list of unacceptable practices, penalties to be assigned, and procedures to be followed in prosecuting cases of alleged academic dishonesty may be found in the Student Handbook.

Students are expected to read and abide by the department's *Academic Practices and Policies*, a copy of which will be distributed with the course syllabus. Unless other instructions are explicitly stated all graded work will be subject to the policy

"Individual Project With Limited Collaboration: In particular, you may receive help from the following persons, in addition to an instructor in this course: any St. Bonaventure University student enrolled in CS 132, and any other person specifically approved by your instructor. You may use the following materials produced by other students: NONE."

In addition, if you do collaborate with anyone other than the instructor, there must be a note to that effect at the top of the solution you turn in.

Academic dishonesty in any form will not be tolerated. Typically the first offense will result in a zero on the assignment. Repeated offenses will likely result in a failing grade for the course. Any offense deemed punishable will also be referred to the Dean of Arts and Sciences.

Services for Students with Disabilities

Students with disabilities who believe that they may need accommodations in this class are encouraged to contact the Disability Support Services Office, Doyle Room 26, at 375-2065 as soon as possible to better ensure that such accommodations are implemented in a timely fashion.

Computer Science Department Goals and Objectives addressed in this course:

Goal 1: Discipline Specific Learning

Students will be able to understand and apply the theoretical tools of computer science to standard problems from the field.

Objectives:

1. Students will learn core concepts of the discipline as determined by a nationally recognized professional computer science education organization.
2. Students will understand and analyze algorithms written in pseudo-code.
3. Students will apply the techniques of asymptotic analysis to blocks of pseudo-code and to program fragments.

Goal 2: Reasoning and Inquiry Skills

Students will be able to read, write, and analyze program fragments and complete programs.

Objectives:

1. Students will write complete programs to solve small problems typical of the field.
2. Students will enhance existing (larger) programs to add capabilities and/or improve the quality of code.
3. Students will design test suites for and run debugging sessions on programs they have written as well as on programs written by others.

Goal 3: Communication Skills

Students will be able gather requirements for a system from third parties, choose a paradigm in which to design the solution, and communicate the parameters of that solution to both professionals in the field and the originating parties.

Objectives:

4. Students will be given the opportunity to present results of their work in oral and written forms; this will include the presentation of posters and/or papers intramurally and extramurally.