CSCI 1321 (Principles of Algorithm Design II), Fall 2007
Syllabus

1 Course description

This course is the second course for computer science majors, following the guidelines established by the Association for Computing Machinery (most recently described in Computing Curricula 2001[1]). It builds on the material of the first course, but with more emphasis on the object-oriented paradigm. This course also partially satisfies common-curriculum requirements (*Understanding the World Through Science* in the pre-2004 requirements, *Understanding Natural Science and Technology* in the current requirements).

Course goals

- Increased ability to solve problems using computer programs of your own construction.
- Ability to use more complex data structures and algorithms in the solution of such problems.
- Basic understanding of how large programming projects can behave and what you can do to make them easier to work with and more flexible.
- Working knowledge of the Java programming language and an IDE (interactive development environment).

Course topics

- Review of control structures, functions, and primitive data types.
- Object-oriented design and programming: encapsulation and information hiding, classes, inheritance, polymorphism.
- Fundamental data structures: arrays, linked lists, stacks, queues, and simple trees.
- Fundamental algorithms: sorting and searching.
- Recursion.
- Introduction to event-driven programming.
- Use of a simple graphics API and GUI construction.
- Basic analysis of algorithms.
- Dynamic memory and dynamically allocated objects.
- Use of standard library classes.

2 Basic information

Class meeting times and location

- TR 12:45pm – 2pm, HAS 340

Prerequisites

- CSCI 1320, or consent of instructor

[1] http://acm.org/education/curric_vols/cc2001.pdf

Instructor contact information

- Dr. Berna Massingill
- Office: HAS 201L
- Office phone: (210) 999-8138
- E-mail: `bmassing@cs.trinity.edu`

Office hours

Scheduled office hours for this semester are as follows.

- Monday 1:30pm – 2:30pm (HAS 201L), 2:30pm – 4:30pm (HAS 340)
- Tuesday 2pm – 4pm (HAS 201L)
- Wednesday 1:30pm – 2:30pm (HAS 201L), 2:30pm – 4:30pm (HAS 340)
- Thursday 2pm – 4pm (HAS 201L)

These times are subject to change; a current schedule will be available on my Web page and outside my office door.

If I'm not at the designated location, I should be somewhere in the building (perhaps in one of the labs helping another student), and there will often be a note on my door saying where to find me.

Notice that some office hours are held in HAS 340: These are "open lab" times, during which I'll be in one of the department's labs, prepared to answer questions. The intent is that students can use these times to work on assignments with someone available to help with any questions or problems.

In addition to scheduled office hours, you're welcome to drop by and see if I'm in my office and free to talk, or you can make an appointment by calling me or sending me e-mail.

E-mail is almost always a good way to reach me; I normally check it fairly often and reply promptly.

3 Course materials

Textbook

- John Lewis and Joseph Chase. *Java Software Structures: Designing and Using Data Structures.* Addison Wesley, second edition, 2005.

Web page

Most course-related information (this syllabus, homework and reading assignments, etc.) will be made available via the Web. The course Web page is a starting point for Web-accessible course material; you can find it linked from my home page (`http://www.cs.trinity.edu/~bmassing`) or directly at `http://www.cs.trinity.edu/~bmassing/Classes/CS1321_2007fall/HTML`.

Other references

There are many, many other books on Java and object-oriented programming and design available; two that may be of interest are listed below. However, there are also many online resources, which together with the textbook will probably be sufficient for this semester and perhaps beyond. In deciding whether to buy additional books you should probably keep in mind that Java continues to evolve, so books become outdated more rapidly than with a more stable language.

- Joshua Bloch. *Effective Java: Programming Language Guide.* Addison Wesley, 2002. Somewhat dated, but full of excellent advice.
- Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software.* Addison-Wesley, 1995. A classic in object-oriented circles.

4 Course requirements

Grading

Grades in this course will be determined by the results of two major exams (a midterm and a final), several in-class quizzes, several homework assignments, and class participation, weighted as follows.

Component	**Maximum points**
Midterm exam	100
Final exam	200
Quizzes	50
Homework	about 500
Class participation	50

Numeric grades will be calculated as a simple percentage, by dividing total points earned on the above components by total points possible. These numeric grades will then be converted to letter grades based on a curve, but in no case will the resulting letter grades be worse than students would receive based on the following scheme.

Numeric grade	**Letter grade**
90 – 100	A-/A
80 – 89	B-/B/B+
70 – 79	C-/C/C+
60 – 69	D/D+
0 – 59	F

Exams

Exams are comprehensive but will emphasize the most recent material. They are scheduled as follows; please plan accordingly (i.e., avoid scheduling anything else for these times).

- Midterm: October 9, in class.
- Final exam: December 12, 8:30 am.

Quizzes

About every other week there will be a short in-class quiz. Dates will be announced via the course Web page. They will usually cover material from recent classes and reading; the questions will be similar in format to those you are likely to see on the major exams. There will be about six quizzes over the course of the semester, and the lowest grade will be dropped.

Homework assignments

Homework (in the form of programming assignments) is a crucial part of this course; much of what you learn will likely be learned in the course of completing the programming assignments. For this course, the assignments will all be based around a single problem. Each assignment will require you to produce a solution to a specific part of the problem so that all can be used together in the end for a single application. Detailed requirements will be provided as part of each assignment; due dates will be announced via the course Web page. You are encouraged to use the department's network of Linux machines, but unless otherwise specified for individual assignments, you may use any other system that provides a suitable environment.

Attendance

Regular class attendance is strongly encouraged; class participation grades will be based in part on attendance.

E-mail

Course-related announcements will sometimes be made by sending e-mail to the Trinity e-mail addresses of all registered students. Students are strongly encouraged to read mail sent to their Trinity addresses frequently.

Late and missed work

Exams can be made up only in cases of documented conflict with a university-sponsored activity, documented medical emergency, or conflict with a religious holiday. Quizzes cannot be made up, but the lowest quiz score will be dropped, so you can miss one quiz without penalty.

Unless otherwise stated for a particular assignment, homework will be accepted up to one class period late, *but no more*, at a penalty of 10 percent off per working day. This penalty may be waived or additional time allowed *at the instructor's discretion* in cases of illness or conflict with a university-sponsored activity or religious holiday.

If you have unusual circumstances (as we all sometimes do), please discuss these with me as far in advance as possible.

Academic integrity at Trinity

All students are covered by a policy that prohibits dishonesty in academic work. The Academic Integrity Policy (AIP)[2] covers all students who entered Trinity before the fall of 2004. The Academic Honor Code[3] covers all those who entered the fall of 2004 or later.

The Integrity Policy and the Code share many features: Each asserts that the academic community is based on honesty and trust; each contains the same violations; each provides for a procedure to determine if a violation has occurred and what the punishment will be; each provides for an appeal process.

The main difference is that the faculty implements the AIP while the Code is implemented by the Academic Honor Council. Under the Integrity Policy, the faculty member determines whether a violation has occurred as well as the punishment for the violation (if any) within certain guidelines. Under the Code, a faculty member will (or a student may) report an alleged violation to the

[2] http://www.trinity.edu/departments/academic_affairs/hb/instrpol/acadintg.htm\#TOP

[3] http://www.trinity.edu/departments/academic_affairs/honor_code/

Academic Honor Council. It is the task of the Council to investigate, adjudicate, and assign a punishment within certain guidelines if a violation has been verified.

Students who are under the Honor Code are required to pledge all written work that is submitted for a grade: "On my honor, I have neither given nor received any unauthorized assistance on this work" and their signature. The pledge may be abbreviated "pledged" with a signature.

Collaboration and academic integrity in this course

Unless otherwise specified, all work submitted for a grade (homework assignments, quizzes, and exams) must represent the student's own individual effort. For students covered by the Academic Honor Code, unless otherwise stated all submitted work will be considered pledged work.

Discussion of homework assignments among students is encouraged, but not to the point where detailed answers are being written collectively. Graded papers and sample solutions from previous years (exams, quizzes, and homeworks) are off limits. Answers that are identical beyond coincidence (either to another student's work or to a sample solution from a previous year) will be considered to be in violation of the AIP or Honor Code, whichever applies, and *will result in appropriate action.* You are responsible for the security of your work, both electronic and hard copy.