EAS 230A **Computer Programming for Engineers** **Fall 2004**

Course Description

EAS230 is an introductory course in computer programming with emphasis on methodology for solving engineering problems. The course will focus on the process for solving engineering problem using the computer, effective use of numerical methods in solving engineering problems, programming using C++ (a modern high level programming language), and visual presentation of solutions. Programming concepts covered include data types, data input and output, control structures, functions and parameter passing, arrays, classes, objects, and pointers. This course extends the computer knowledge developed initially in EAS 140. Major course modules will address:

- Understanding, analyzing, and solving a scientific and engineering word problem using a step-wise refinement process.
- Introduction to C++ language and problem solving and programming with C++ and numerical methods.
- Applying numerical methods to representative engineering problems and handling large volumes of data stored in matrices.
- Presentation of solutions to the problems using graphical and visual methods.

Hands on practical projects using Microsoft Visual C++ 6.0 will support concepts discussed during the lecture.

On completion of this course students will be able analyze an engineering problem, design and develop a computer program to solve the problem, test the validity of the computerized solution, and present the completed solution in a user-friendly format.

Course Information

The course is delivered in two 50-minutes lectures and one 50-minutes laboratory per week.

Newsgroup:	sunyab.eas.230
Website:	http://www.cse.buffalo.edu/~bina/EAS230/fall2004/index.html

A Section:

Instructor:	Bina Ramamurthy (bina@cse.buffalo.edu) (bina)
Lecture Time:	MW 5.00-5.50PM
Lecture Location:	112 Norton
Bina's Office:	127 Bell Hall
Office Hours:	Tue: 9:00 – 11:00AM, Wed: 10:00-11:00AM

B Section:

Instructor:	Tim Terrill
Lecture Time:	TT: 8.30-9.20AM
Lecture Location:	110 Knox
Tim's Office:	124 Bell Hall
Office Hours:	Mon: 10:00 – 11:30AM, Thu: 9:30 – 11:00AM

All the laboratory sessions will be held in Furnas 211. You are required to attend the lab you are registered for.

Laboratory	Day	Time	TA Name
A1	Wed	3.00-3.50PM	
A2	Wed	6.00-6.50PM	
A3	Thu	11.00-11.50AM	
A4	Fri	9.00-9.50AM	
B1	Mon	3.00-3.50PM	
B2	Wed	4.00-4.50PM	
B3	Fri	12.00-12.50PM	
B4	Fri	5.00-5.50PM	

Textbook

Engineering Problem Solving with C++ by Dolores M. Etter and Jeanine A. Ingber, ISBN 0-13-091266-2, Prentice-Hall Inc., 2003.

References

Links for Visual C++ references (language and Visual Studio 6.0) will be available on the course website.

Pre-requisites

The pre-requisite for this course is ***EAS140 or an equivalent course that teaches computer literacy and basic operation of a computer.***

Grading Distribution

Grades will consist of the following components:

Component (Quantity)	**Percentage**
Programming assignments (8)	60%
Midterm (1)	15%
Final (1)	25%

here will be 7 lab /programming exercises assigned at approximately one every two weeks. You will prepare a portfolio (electronic or hardcopy) of all the labs exercises (from problem statement to final solution and testing details). Eighth lab will introduce you to the broader application areas of engineering problem solving. More details about this lab (lab 8) will be given to you after mid-semester exam.

Point distribution guideline will be as follows:

Point Range	Letter Grade
95.00-100	A
90.00-94.99	A-
85.00-89.99	B+
80.00-84.99	B
75.00-79.99	B-
70.00-74.99	C+
65.00-69.99	C
60.00-64.99	C-
55.00-59.99	D+
50.00-54.99	D
0-49.99	F

We reserve the right to alter component weighting or provide a "curve" on an assignment as warranted.

Labs

Lab assignments constitute a major portion of the course. Over the semester, you will be given eight lab assignments. The assignments will require you to practice problem solving skills, work with the C++ programming language, design and implement computer solutions to engineering and scientific problems.

You will be given approximately 2 weeks to complete each lab exercise. *Do not be lulled into a safe sense of security. Do not think you have a lot of time to implement each lab!* Although many parts of the lab assignment only require a few lines of code to implement, each lab does require that you have a strong understanding of the fundamental concepts you learn during lectures. This understanding takes time, patience, and an experimenting attitude.

You are required attend the recitation/lab you are registered for. This will be held at Furnas 211 that has computing facilities and work environment for you to work on the lab. A teaching assistant (TA) will conduct the lab/recitation sessions. He/she will explain the problem described in the lab exercise assigned to you and will help with hints on solving the problem.

Lab Assignment Protocol:

1. Lab description will be handed out and explained during lecture. Simultaneously it will be available on the course web page.
2. TA assigned to the lab session will further elaborate the details of the lab exercises during recitations.
3. You will work on the lab exercises during the recitation/lab time and during extra open lab hours and at your own time in the public engineering labs or on your own computer.
4. After completing the program and testing it successfully you will demo the work for your TA on or before the due date.

5. On the due date you will submit online the program source code and documentation specified in the lab handout.

The TA will also run test examples against your code to check your solution's overall correctness. The TA will provide demo times. It is your responsibility to demo your project, or you will receive a zero for that portion of the grade.

Exams

There will be a midterm that will be administered and graded before the resign date. Midterm material will cover all lecture and reading assignments before the exam, as well as concepts from the lab assignments. The final is a comprehensive exam, covering all lecture, lab, and homework areas. Exams are closed book, closed notes, and closed neighbor. However you will be allowed one 8.5 x 11 sheet of information, hand-written by you.

Attendance Policy

Although attendance is not officially required, you are still responsible for the contents of all lectures and recitations (your assigned section). If you know that you are going to miss a lecture or a recitation, have a ***reliable*** friend take notes for you. Of course, there is no excuse for missing due dates or exam days. A Detailed lecture schedule is enclosed. Recitations are designed to review difficult concepts in the class and to spend additional time discussing the lab work required for the course. The recitation is your time to communicate with your TA about the course. Use the opportunity to the fullest.

Grading Policy

All assignments will be graded and returned in a timely manner. When an assignment is returned, you will have a period of one week to contest any portion of the grade. The TA who graded your assignment will be the first person to resolve a grading conflict. If the conflict cannot be resolved, the instructor will mediate the dispute. The judgment of the instructor will be final in all such cases. When contesting a grade, you must be able to demonstrate how your particular solution is correct. Also, when contesting a grade, the instructor or TA reserves the right to re-evaluate the entire lab or exam, not just the portion in dispute.

Incomplete Policy

We only grant incompletes in this course under the direst of circumstances. Incompletes are not designed as stalling tactic to defer a poor performance in a class.

Academic Integrity Policy

UB's definition of Academic Integrity in part is, "Students are responsible for the honest completion and representation of their work". It is required as part of this course that you read and understand the departmental academic integrity policy. There is a very fine line separating conversation pertaining to concepts and academic dishonesty. You are allowed to converse about general concepts, but in no way are you allowed to share code or have one person do the work for others. You must abide by the UB and Departmental Academic Integrity policy at all times. **If a student is caught violating the academic integrity policy, he/she will minimally receive a ZERO in that lab/exam. Repeated violation may result in F grade for the student for the course.**

Web Site and Newsgroup

The course website should be checked frequently for important news. Course assignments, slides, grade reporting, and general hints and tips will be posted on the website. The newsgroup will be used as a method to ask questions regarding assignments. We encourage all students to participate in newsgroup conversations. Generally, we will try to answer newsgroup questions within a period of 48 hours of their posting.

Students with Disabilities

If you have special needs due to a disability, you must be registered with the Office of Disability Services. If you are registered with ODS, I need to know as soon as possible.

Tentative Lesson Plan:

Date	Topics Covered	Reading Assignment
Aug.30	Welcome. Course outline; Goals and objectives; Rules and responsibilities;	Ch.1
Sep.1	Introduction to Lab environment: Visual C++ environment; Sample commands demo. Lab 1 assigned.	Handout
Sep.8	Problem solving steps: real world problem, model and word description, analysis, solution design and representation; Algorithmic problem solving.	Handout
Sep.13	C++ program structure; Simple data representation: variables and constants; arithmetic operators.	Ch.2
Sep.15	Standard input/output; Complete program with input and output; Lab 2 assigned.	Ch.2
Sep.20	Numerical techniques: linear interpolation; Applied problem solving: wind tunnel analysis.	Ch.2
Sep.22	Numerical problem solving (contd.): Math library functions; Applied problem solving: Velocity computation; Lab 3 assigned.	Ch.2
Sep.27	Algorithm development; control structures; Conditional expressions; Pseudo code and flowchart design representation;	Ch.3
Sep.29	Selection statements: if..else..then; Multiway selection: switch statement;	Ch. 3
Oct.4	Repetition statements: while, do/while; for loop; Problem solving: weather balloon tracking. Lab 4 assigned.	Ch. 3
Oct.6	Exercises with selection and repetition control structures.	Classnotes
Oct.11	Review for mid-term exam	
Oct.13	Mid-term exam	Ch.1-3
Oct.18	Working with data: defining file streams, load from and store into files; Problem solving: data filters; Lab 5 assigned.	Ch. 4
Oct.20	Modular design; Functional decomposition of problems; C++ functions.	Ch.5
Oct.25	Problem solving using simple functions;	Ch. 5
Oct.27	Parameter passing; Built-in functions and libraries;	Ch. 5
Nov.1	Recursion and problem solving using recursive functions.	Ch. 5
Nov.3	Random number generation and use of random numbers in problem solving. Lab 6 assigned.	Ch. 5
Nov.8	Data structures: One dimensional arrays.	Ch. 6
Nov.10	Sorting and searching algorithms.	Ch. 6
Nov.15	Problem solving using arrays. Lab 7 assigned.	Ch. 6
Nov.22	String and Vector class; using classes and objects.	Ch. 6
Nov.29	Matrices and two dimensional arrays.	Ch. 7
Dec.1	Problem solving using matrices.	Ch. 7
Dec.6	Complete problem solving process: review	
Dec.8	Review for final exam	