**CS 335** **COMPUTER ARCHITECTURE** **SPRING 2004**

Syllabus & Course Policies

INSTRUCTOR: Robert England EMAIL: englandr@rhodes.edu
OFFICE: OH 419 PHONE: 843-3725

OFFICE HOURS:

2:00-3:00pm, MWF
I'm available lots of other times. Please make an appointment!

TEXT:

*Computer Organization & Design: The Hardware/Software Interface,*
by Patterson & Hennessy.
Supplemental material will be distributed in class as needed.

COURSE DESCRIPTION:

This course focuses on the five basic components of a computer --- datapath, control, memory, input, and output --- but in greater depth and detail than, say, a Computer Organization course or a Systems Programming course. We will also explore the fundamentals of the logic implementation and execution of several types of instructions, as well as the relationship between higher-level languages and machine language.

PREREQUISITES:

CS 231 and CS 241.

WITHDRAWALS:

The last day to withdraw from this class is Friday, 29 Oct 2004.

TOPICS: [bluntly stated, and not necessarily in this order]

Intensive review (?) of digital logic, internal data representation, and assembly language
Low-level implementation of subroutines
ALU design
- Basic processor design
- Basic pipelining concepts

Memory hierarchy design
Input / Output
Parallel processing

Mapped to the book, we'll concentrate on chapters 3 through 8.

GRADING:

| | | |
|---|---|---|
| Program assignments and presentations: | 50% | (10% for attendance) |
| 2 in-class tests: Tues 28 Sept, Tues 9 Nov: | 30% | |
| Final exam: 1:00pm, Weds 15 Dec: | 20% | |

ATTENDANCE:

Attendance will be checked each class period. The format of this course will allow for regular student participation through presentations of assigned readings and the follow-up discussions of these presentations. Your participation each class meeting is important regardless of whether or not you are a presenter. You will get ten free points (our of one hundred) toward your final grade for this course if you miss class fewer than three times. You must be present on any day you are scheduled to make a presentation in order to receive credit for the presentation. (No make-ups on those.)

Students are responsible for all material assigned or covered in class. The instructor's own lecture notes will not be made available for copying or review.

TESTS:

Both of the regular tests and the exam will be closed book, closed notes. No make-up tests will be given unless prior arrangements are made with the instructor.

PROGRAM ASSIGNMENTS:

The nature of the material in this course is such that programming assignments may take a variety of forms --- for example, assembler code, digital logic designs, regular C code, etc. Each program will be assigned a letter grade and carry a weight approximately twice that of a single in-class presentation.

Programming assignments must be turned in on the due date to receive full credit. Programs will be accepted late, but with a penalty of one letter grade per day. You get a three late day allowance.

PRESENTATIONS:

In contrast to the standard lecture format, in this course, each student will regularly have the opportunity to make an in-class presentation of material covered, typically leading into an in-class discussion during the latter part of a class period. This material may be a section of a chapter in the book given as a reading assignment, or a small set of homework problems, or a solution to a programming assignment, etc. You will know at least one class period in advance of each presentation what material you are responsible for presenting. Be prepared to answer questions about your presentations and to lead the follow-up discussions.

Individual presentations will be graded as Outstanding (100 pts), High Pass (95 pts), Pass (85 pts), Low Pass (70 pts), and Fail (50 pts). If you are absent when it's your turn to present, you get nothing (0 pts).

ACADEMIC INTEGRITY:

All programs and tests must be the student's *own* work. Copying all or part of an assignment, or downloading code from the Internet and submitting it as your own, or having someone else write code for your assignment, or having someone else debug your assignment, or *allowing* someone else to copy from you, or coding or debugging someone else's assignment --- these are all included in the definition of reportable Honor Code violations for this course. If you have any doubts about whether or not a program development practice on an assignment is acceptable, clear it with the instructor before proceeding.

The instructor reserves the right to alter this syllabus as necessary.