Computer Science Department
cs.salemstate.edu

CSC 295 Computer Architecture and Organization **3 cr.**

Instructor: TBA **Office:** location **Phone:** (978) 542-extension
email: TBA@salemstate.edu **Office Hours:** days and times

Section	Time	Room	Final Exam
nn	days and times	location	date and time

Catalog description:

This course examines the basic principles of computer systems and how these concepts relate to the design of such systems. Both hardware and software concepts and the interdependence between them are dealt with. The determination of basic trade-offs and the related decisions are discussed. Logic level designs, data representations, computer circuits, fundamental computer operations, program creation, I/O programming, processing elements, links and interfaces, memory hierarchy, and memory management are covered. Three lecture hours per week.

Prerequisites: **CSC 105 and CSC 115.**

Goals:

The goals of this course are to introduce students to the concepts of the organization and architecture of computer systems, from the physical and logic levels through the intermediate levels to the higher-level-language level, and the methodologies and problem-solving strategies used define and implement the necessary ingredients. Specific goals are to:

CG01: present the concept of a computer system as a series of levels, each with its own properties and methodologies;
CG02: introduce a series of problem-solving methodologies relating to the various system levels;
CG03: discuss problem-solving techniques based on the presented methodologies.

Objectives:

Upon successful completion of the course, a student will have:

CO01: demonstrated knowledge of the concepts of computer architecture and organization;
CO02: demonstrated the ability to apply appropriate problem-solving strategies to solve a selection of typical problems in computer architecture and organization;
CO03: gained hands-on experience designing an ALU based on a set of specified requirements;
CO04: implemented a simple instruction set computer with a control unit and a data path;
CO05: developed a good understanding of memory hierarchy in a computer system;
CO06: developed an understanding of an I/O subsystem supporting processor programmed I/O, direct memory access and interrupt structures;
CO07: developed an understanding of basic concepts of a multi-core processor design;
CO08: participated in at least one group project that involves solution design, analysis and evaluation.

Student Outcome vs. Course Objectives matrix

Student Outcome (condensed form)	**CO01**	**CO02**	**CO03**	**CO04**	**CO05**	**CO06**	**CO07**	**CO08**
SO-1	✓	✓	✓	✓	✓	✓	✓	
SO-2		✓	✓	✓	✓	✓	✓	
SO-3								✓
SO-4							✓	
SO-5								✓
SO-6	✓	✓	✓	✓	✓	✓	✓	

Note:

SO-1 Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.

SO-2 Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

SO-3 Communicate effectively in a variety of professional contexts.

SO-4 Recognize professional responsibilities and make informed judgements in computing practice based on legal and ethical principles.

SO-5 Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.

SO-6 Apply computer science theory and software development fundamentals to produce computing-based solutions.

Topics:

- Computers and their applications **PL2(1)**
- Data representations **AR2(4)**
  - text
  - numeric data types
  - evaluation of expressions
- Logic level design **AR1(3)**
  - elementary logic gates
  - combinational logic design
  - elementary sequential circuits
- Computer circuits **AR1(4)**
  - electrical properties
  - combinational logic implementations
  - important combinational circuits
  - sequential circuits
  - link connections
  - integrated circuits and technologies
- Fundamental computer operations **AR3(3), AR6(2)**
  - machine language and assembler language instructions
  - stacks
  - procedures
  - macros
  - instruction execution time
- Program creation
  - assemblers
  - compilers
  - linking and address adjustment
  - loading and address adjustment
- Input/output programming **AR5(1.5)**
  - programmed I/O
  - interrupt I/O
  - direct memory access
  - I/O elements
- Processing elements **AR3(8), AR6(3)**
  - macroinstruction execution
  - internal bus transfers
  - detailed internal architecture example
  - microcontrol
  - reduced instruction set computers (RISC)
  - packaging
- Links and interfaces **AR5(1.5)**
  - system buses
  - interfaces
  - data links
- Memory hierarchy **AR4(2.5)**
  - mass storage
  - main memory
  - multiple-port memory

- cache memory
- hierarchy design

- Memory management **AR4(2.5)**
  - mass storage management
  - main memory management
  - memory management hardware
  - virtual memory
- Operating systems **AR6(2)**
  - uniprogramming systems
  - multiprogramming systems
  - organization of a multiprogramming system
  - sharing resources
- Parallel processing **AR7(3)**
  - multiprocessing
  - pipelining
  - vector and matrix processing
  - high-performance computing

This course provides an implementation-independent treatment of the subject, emphasizing general and widely applicable principles rather than focusing on implementation methods, which may be specific to a particular type or model of computer. To give students hands-on experiences, laboratory activities are used in the course. Among these laboratory activities, at least 2 require students to work in teams to accomplish larger-scale assignments.

The course grade will be determined using the following approximate weights: final examination - 25%, midterm examination - 25%, laboratory activities - 25%, other tests, quizzes, and written homework - 25%.

Course Objective / Assessment Mechanism matrix

	Homework	Exams	Laboratory Activities
CO01	✓	✓	✓
CO02	✓	✓	✓
CO03	✓		✓
CO04			✓
CO05	✓	✓	✓
CO06	✓	✓	
CO07	✓	✓	
CO08			✓

Bibliography:

Baer, Jean-Loup. **Microprocessor Architecture: From Simple Pipelines to Chip Multiprocessors**, Cambridge University Press, 2010.

Bryant, Randal E.; O'Hallaron, David R. **Computer Systems: A Programmer's Perspective**, **Third Edition**, Addison-Wesley, 2016.

Hamacher, Carl; Vranesic, Zvonko; Zaky, Safwat; Manjikian, Naraig. **Computer Organization and Embedded Systems**, Sixth **Edition**, McGraw-Hill, 2012.

Harris, David; Harris Sarah **Digital Design and Computer Architecture**, **Second Edition**, Morgan Kaufmann, 2012.

Hennessy, David; Patterson, John. **Computer Organization and Design: The Hardware/Software Interface. Fifth Edition.** Morgan Kaufmann, 2014.

Heuring, Vincent P.; Jordan, Harry F. **Computer Systems Design and Architecture**, **Second Edition**, Prentice Hall, 2004.
Jin, Lan; Hatfield, Bo, **Computer Organization: Principles, Analysis & Design**, **First Edition**, Tsinghua University Press, 2004.
Marcovitz, Alan B. **Introduction to Logic and Computer Design, Third Edition**, McGraw Hill, 2008
Ramachandran, Umakishore; Leahy, William D. Jr. **Computer Systems: An Integrated Approach to Architecture and Operating Systems**, **First Edition**, Addison Wesley, 2011.
Shen, John Paul, Lipasti, Mikko H. **Modern Processor Design: Fundamentals of Superscalar Processors**, **First Edition**, McGarw Hill, 2005.
Stallings, William. **Computer Organization and Architecture. Eighth Edition.** Prentice-Hall, 2010.
Stallings, William. **Computer Organization and Architecture Designing for Performance. Tenth Edition.** Prentice-Hall, 2016.
Tanenbaum, Andrew. **Structured Computer Organization. Fifth Edition.** Prentice-Hall, 2005.
Williams, Rob. **Computer Systems Architecture – A Networking Approach**, **Second Edition**, Pearson Prentice Hall, 2006

Academic Integrity Statement:

"Salem State University assumes that all students come to the University with serious educational intent and expects them to be mature, responsible individuals who will exhibit high standards of honesty and personal conduct in their academic life. All forms of academic dishonesty are considered to be serious offences against the University community. The University will apply sanctions when student conduct interferes with the University primary responsibility of ensuring its educational objectives." Consult the University catalog for further details on Academic Integrity Regulations and, in particular, the University definition of academic dishonesty.

The Academic Integrity Policy and Regulations can be found in the University Catalog and on the University website (http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the University's Policy* and will be dealt with according to the University's formal Procedures. *Copying without attribution is considered cheating in an academic environment - simply put,* ***do not do it!***

University-Declared Critical Emergency Statement:

In the event of a university-declared emergency, Salem State University reserves the right to alter this course plan. Students should refer to www.salemstate.edu for further information and updates. The course attendance policy stays in effect until there is a university-declared critical emergency.

In the event of an emergency, please refer to the alternative educational plans for this course, which will be distributed via standing class communication protocols. Students should review the plans and act accordingly. Any required material that may be necessary will have been previously distributed to students electronically or will be made available as needed via email and/or Internet access.

Equal Access Statement:

"Salem State University is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. **Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately.** Students with Disabilities who have not previously done so should provide documentation to and schedule an appointment with the Office for Students with Disabilities and obtain appropriate services."

Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via email.