CSC 295  Computer Architecture and Organization  3 cr.

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 are considered and the interdependence between them are covered. Determining basic tradeoffs and related decisions are covered. 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.
Prerequisite:  CSC 202J with grade of C+ or higher; CSC 215, PHS 205.

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 present:

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

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

CO01:  demonstrated knowledge of the concepts of computer architecture and organization;
CO02:  practiced the use of various problem solving strategies;
CO03:  demonstrated the ability to apply appropriate problem-solving strategies to solve a selection of typical problems in computer architecture and organization.

<table>
<thead>
<tr>
<th></th>
<th>PO1</th>
<th>PO2</th>
<th>PO3</th>
<th>PO4</th>
<th>PO5</th>
<th>PO6</th>
<th>PO7</th>
<th>PO8</th>
<th>PO9</th>
<th>PO10</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO01</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO02</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CO03</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Topics:
- Computers and their applications
  - hardware and software
- Data representations
  - text
  - numeric data types
    - non-negative integers
    - signed integers
    - range and scaling
    - real numbers
  - evaluation of expressions
- Logic level design
  - elementary logic gates
  - combinational logic design
  - elementary sequential circuits
- Computer circuits
  - electrical properties
  - combinational logic implementations
  - important combinational circuits
  - sequential circuits
  - link connections
• integrated circuits and technologies

Fundamental computer operations
  • 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
  • programmed I/O
  • direct memory access
  • I/O elements

Processing elements
  • macroinstruction execution
  • internal bus transfers
  • detailed internal architecture example
  • microcontrol
  • reduced instruction set computers (RISC)
  • packaging

Links and interfaces
  • system buses
  • interfaces
  • data links

Memory hierarchy
  • mass storage
  • main memory
  • multiple-port memory
  • cache memory
  • hierarchy design

Memory management
  • mass storage management
  • main memory management
  • memory management hardware
  • virtual memory

Operating systems
  • uniprogramming systems
  • multiprogramming systems
  • organization of a multiprogramming system
  • sharing resources

Parallel processing
  • 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.

The course grade will be determined using the following approximate weights: final examination - 35%, midterm examination - 35%, other tests, quizzes, and written homework - 30%.
<table>
<thead>
<tr>
<th></th>
<th>Homework</th>
<th>Exams</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO01</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>CO02</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>CO03</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

Bibliography:


