CSC 225 – Introduction to Computer Organization

1. CSC 225 – Introduction to Computer Organization

2. credit units 4 contact hours 6

3. Course Coordinator: Julie Workman

4. Textbook (or other required material):
   Introduction to Computing Systems – from bits & gates to C & beyond or similar material

5. a. Course Description:
   Introduction to computer systems. Simple instruction set architecture and the computer hardware needed to implement that architecture. Machine and assembly language programming. 3 lectures, 1 laboratory.

   b. Prerequisite: CSC/CPE 202.

   c. Required/Elective/Selective Elective for CPE, CSC, EE, SE

<table>
<thead>
<tr>
<th>Required</th>
<th>CSC</th>
<th>CPE</th>
<th>SE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elective</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Selective Elective</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6. a. Course Learning Objectives
   The student will be able to:
   • Explain how computers work from the gates to the basic instructions used to run the programs.
   • Explain how high-level programs look to the computer and discuss high-level program design as relates to the assembly level.
   • Translate a high-level instruction into assembly language
   • Design and implement a 200-line program in assembly language
   • Translate an assembly language instruction into its machine language equivalent

   b. Level at which Student Outcomes are addressed
   (“B” = Basic level, “I” = Intermediate level, “A” = Advanced level)

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSC</td>
<td>B</td>
<td>B</td>
<td></td>
<td>B</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>SE/</td>
<td>B</td>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7. Major Topics Covered: (number of lecture hours per)
   • Von Neumann computation model – 2 hours
   • LC-3 ISA and hardware – 4 hours
• Assembly programming and mnemonics – 3 hours
• Assembly programming – 3 hours
• I/O, interrupts, and exceptions – 3 hours
• The stack – 3 hours
• Control structures in C / mapping to assembly – 3 hours.
• Traps and subroutines – 2 hours
• Pointers and arrays in C and assembly – 3 hours
• Functions in C, calling convention in assembly – 3 hours
• Data structures in C and assembly – 3 hours