**INTI INTERNATIONAL UNIVERSITY**
**COURSE STRUCTURE**

**PROGRAMME: DIPLOMA IN INFORMATION COMMUNICATION TECHNOLOGY**

<table>
<tr><td>1.</td><td colspan="8">NAME OF COURSE/MODULE : COMPUTER ORGANIZATION</td></tr>
<tr><td>2.</td><td colspan="8">COURSE CODE: ICT2101</td></tr>
<tr><td>3.</td><td colspan="8">RATIONALE FOR THE INCLUSION OF THE COURSE/MODULE IN THE PROGRAMME :<br>Students need to be exposed to the fundamentals of computer systems organisation and architecture as well as components design. Students are also expected to be able to design a simple processor, instruction set and addressing, control structures, memory management as well as interrupts and I/O structures.</td></tr>
<tr><td rowspan="3">4.</td><td rowspan="2">STUDENT LEARNING TIME (SLT)</td><td colspan="5">Total Face to Face</td><td colspan="2">Total Student Independent Learning Time</td></tr>
<tr><td>L</td><td>T</td><td>P</td><td>O</td><td>A</td><td>OL</td><td>IL</td></tr>
<tr><td>L = Lecture<br>T = Tutorial<br>P = Practical<br>O= Others<br>A= Assessment<br>OL=Online learning<br>IL= Independent learning</td><td>28</td><td>0</td><td>28</td><td>0</td><td>3</td><td>14</td><td>87</td></tr>
<tr><td>5.</td><td colspan="8">CREDIT VALUE: 4</td></tr>
<tr><td>6.</td><td colspan="8">PREREQUISITE (if any): ICT1103 : Structured Programming</td></tr>
<tr><td>7.</td><td colspan="8">LEARNING OUTCOMES:<br>On completion of the course, students will be able to:<br>1. Describe the architecture of N-bit microprocessor and the corresponding memory and I/O subsystem.<br>2. Write and assemble assembly language programs and run them to the target microprocessor.<br>3. Explain the memory organisation, I/O organisation, CISC vs RISC architecture of a computer system.</td></tr>
<tr><td>8.</td><td colspan="8">SYNOPSIS:<br>This course covers the structure of computer system with concentration in the architecture of microprocessor, memory sub-system, I/O sub-system, and Assembly language. It assumes some programming experience and equips students with knowledge of computer structure, operation and input/output facilities.</td></tr>
<tr><td>9.</td><td colspan="8">MODE OF DELIVERY: Lectures, Laboratory, Tutorials.<br>Lecture, group discussion and tutorials are conducted both face to face and online.</td></tr>
<tr><td>10.</td><td colspan="8">ASSESSMENT METHODS AND TYPES:<br><br>
<table>
<tr><th>Method</th><th>Types</th><th>Weightage (%)</th></tr>
<tr><td rowspan="2">Continuous assessment</td><td>Test</td><td>20</td></tr>
<tr><td>Assignments</td><td>40</td></tr>
<tr><td>Summative Assessment</td><td>Final Examination</td><td>40</td></tr>
</table></td></tr>
</table>

11. **CONTENT OUTLINE OF THE COURSE/MODULE AND THE SLT PER TOPIC:**

| Sessions | Topics | LO | L | T | P | OL | Total | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | O | A | IL |
| 1-2 | **Introduction to Computer Technology**<br>General definition of a computer system. Technological basis for such computers. The history of digital computer system. | 1 | 2 | 0 | 2 | 1 | | | |
| 3-4 | **Number System and Data Representation**<br>Introduction to number systems: Binary, Octave, Hexadecimal, and Decimal. Data representation in computer: signed & unsigned numbers, real numbers, BCD (Binary Coded Decimal), and ASCII code. | 1 | 2 | 0 | 2 | 1 | | | |
| 5-6 | **N-bit microprocessors.**<br>Concept of stored program computer. Brief history of N-bit microprocessor development. Architecture of an actual microprocessor (8086). System buses. Fetch and execute cycle. Register Sets. | 2 | 2 | 0 | 2 | 1 | | | |
| 7-8 | **8086 Programming Model**<br>The concept of segmented memory. Addressing Modes. Instruction set architecture (ISA). | 2 | 2 | 0 | 2 | 1 | | | |
| 9-16 | **8086 Instruction Set and Assembly Language Programming**<br>Various type of instructions including data transfer instructions, arithmetic instructions, logical instructions, branch instructions, PUSH & POP instructions, and flags control instructions. Assembly language programming in Turbo Assembler environment. DOS and BIOS services for I/O control. Procedures call. | 4 | 8 | 0 | 8 | 4 | | | |
| 17-20 | **Memory Organisation**<br>Memory hierarchy. Cache memory. Memory management in real mode and protected mode. Segmented memory and memory paging. Virtual Memory. | 3 | 4 | 0 | 4 | 2 | | | |
| 21-24 | **I/O Organisation**<br>I/O address mapping. I/O programming. CPU – I/O interaction: polling, interrupt-driven, DMA. | 3 | 4 | 0 | 4 | 2 | | | |
| 25-28 | **CISC vs RISC architecture.**<br>32-bit and 64-bit microprocessor. Architectures that helps to improve CPU performance: pipelining, superscalar, branch prediction, hyperthreading.<br><br>**Final Examination** | 3 | 4 | 0 | 4 | 2 | | | |
| | TOTAL | | 28 | 0 | 28 | 14 | | 3 | 87 |

*Lecture (L), Tutorial (T), Practical (P), Other (O), Assessment (A), Online Learning (OL); Independent Learning (IL); Learning Outcome (LO)*

12. **MAIN REFERENCE(S) SUPPORTING COURSE:**

- Patterson D.A., Hennessy J.L. (2011), Computer Organization and Design, Fourth Edition: The Hardware/Software Interface, Elsevier Science, ISBN: 0123747503, ISBN-13: 978-0123747501

**ADDITIONAL REFERENCES**:

- Irive K.R. (2014), Assembly Language for x86 Processors, 7th Edition, Prentice Hall. ISBN: 978-0133769401
- Stallings W., (2012), Computer Organization and Architecture Designing for Performance, 9th edition, Prentice Hall, New Jersey. **ISBN-10:** 013293633X, **ISBN-13:** 978-0132936330

13. **OTHER ADDITIONAL INFORMATION (if any):**

**Final Examination Format**
Duration: 2 hours
Section A: Answer TWO compulsory questions.
Section B: Answer any TWO out of THREE questions.
All questions carry equal marks

**Grading scale**
A+ (90-100), A (80-89), A- (75-79), B+ (70-74), B (65-69), B- (60-64), C+ (55-59), C (50-54), C- (45-49), D (40-44), F (0-39).
Resit Pass (50-100), Resit Fail (0-49).

**Laboratory Work Specifications (if any)**

| **Week** | **Practical Work** |
|---|---|
| 1-2 | DOS Debug and low level instructions. |
| 3-4 | Analysis of internal microprocessor's registers and addressing modes. |
| 5-8 | Advanced usage of MOV in MASM |
| 9-10 | Write assembly language programs using access to register and stack |
| 11-12 | Write assembly language programs using branching instructions |
| 13-14 | Write assembly language programs using subroutines |

**Important Note:**
**A student who obtains a grade C- (45 -49 marks) in a 100% coursework module is required to resubmit the coursework component determined by the lecturer and ascertained at the Exam Board. Resubmission marks will be capped at a maximum of 50 marks or a grade C.**

**A passing mark can only be achieved when the student attempts both the coursework and final exams.**