Course Title	**Computer Organization & Assembly Language**
Course Code	**DC-221**
Credit Hours	3
Category	Domain Core
Prerequisite	None
Co-Requisite	None
Follow Up	None
Course Description	Introduction to computer systems: Information is bits + context, programs are translated by other programs into different forms, it pays to understand how compilation systems work, processors read and interpret instructions stored in memory, caches matter, storage devices form a hierarchy, the operating system manages the hardware, systems communicate with other systems using networks; Representing and manipulating information: information storage, integer representations, integer arithmetic, floating point; Machine-level representation of programs: a historical perspective, program encodings, data formats, accessing information, arithmetic and logical operations, control, procedures, array allocation and access, heterogeneous data structures, putting it together: understanding pointers, life in the real world: using the gdb debugger, out-of-bounds memory references and buffer overflow, x86-64: extending ia32 to 64 bits, machine-level representations of floating-point programs; Processor architecture: the Y86 instruction set architecture, logic design and the Hardware Control Language (HCL), sequential Y86 implementations, general principles of pipelining, pipelined Y86 implementations
Text Book(s)	Computer Systems: A Programmer's Perspective, 3/E (CS:APP3e), Randal E. Bryant and David R.O' Hallaron, Carnegie Mellon University
Reference Material	Robert Britton, MIPS Assembly Language Programming, Latest Edition, Computer System Architecture, M. Morris Mano, Latest Edition, Assembly Language Programming for Intel- Computer, Latest Edition