Computer Science Courses for Undergraduate Programme of study with Computer Science discipline as one of the two Core Disciplines
(For e.g. courses for B.A. Programmes with Computer Science as Non-major Discipline)

CREDIT DISTRIBUTION, ELIGIBILITY AND PRE-REQUISITES OF THE COURSE

Course title & Code	Credits	Credit distribution of the course			Eligibility criteria	Pre-requisite of the course (if any)
		Lecture	Tutorial	Practical/ Practice		
DSC03: Computer System Architecture	**4**	**3**	**0**	**1**	**Passed 12th class with Mathematics**	NIL

Learning Objectives

This course introduces students to the fundamental concepts of digital computer organization, design, and architecture. It aims to develop a basic understanding of the building blocks of a computer system and highlights how these blocks are organized together to architect a digital computer system.

Learning outcomes

On successful completion of the course, students will be able to:

- Design combinatorial circuits using basic building blocks. Simplify these circuits using Boolean algebra and Karnaugh maps. Differentiate between combinational circuits and sequential circuits.
- Represent data in binary form, convert numeric data between different number systems, and perform arithmetic operations in binary.
- Determine various stages of the instruction cycle and describe interrupts and their handling.
- Explain how the CPU communicates with memory and I/O devices.
- Simulate the design of a basic computer using a software tool.

SYLLABUS OF DSC-3

Unit 1 (9 hours)

Digital Logic Circuits: Digital Logic Gates, Flip flops and their characteristic table, Logic circuit simplification using Boolean algebra and Karnaugh map, Don't care conditions, Combinational circuits, Introduction to Sequential Circuits

Unit 2 (7 hours)

Digital Components: Decoders, Encoders, Multiplexers, Binary Adder, Binary Adder Subtractor, Binary Incrementor, Registers, and Memory Units

Unit 3 (13 hours)

Data Representation: Binary representation of both numeric and alphanumeric data, representation of numeric data in different number systems, (Binary, Octal, Decimal and Hexadecimal), conversion from one number system to another, complements, representation of signed and unsigned numbers, addition and subtraction of signed and unsigned numbers and overflow detection.

Unit 4 (9 hours)

Basic Computer Organization and Design: Stored program organization, Computer registers, Instruction set and their completeness, Instruction cycle, Memory reference instructions, Register reference instructions, Input- Output reference instructions, Interrupt cycle, Addressing modes.

Unit 5 (7 hours)

Input-Output Organization: I/O interface, I/O vs. Memory Bus, Isolated I/O, Memory Mapped I/O, Direct Memory Access.

Essential/recommended readings

1. M. Morris Mano, *Computer System Architecture*, 3rd edition, Pearson Education, 2017.
2. Linda Null, Julia Lobur, *Essentials of Computer Organization and Architecture*, 5th Edition, 2019.

Additional References

3. D. Comer, *Essentials of Computer Architecture,* 2nd edition, CRC Press, 2017.

Suggested Practical List (If any): (30 Hours)

Practical exercises such as

(Use Simulator – CPU Sim 3.6.9 or any higher version for the implementation)

1. Create a machine based on the following architecture:

Registers

IR	DR	AC	AR	PC	I	E
16 bits	16 bits	16 bits	12 bits	12 bits	1 bit	1 bit

Memory 4096 words 16 bits per word	Instruction format 15 12 11 0

	Opcode	Address

Basic Computer Instructions

Memory Reference			Register Reference	
Symbol	Hex		Symbol	Hex
AND	0xxx	Direct Addressing	CLA	7800
ADD	1xxx		CLE	7400
LDA	2xxx		CMA	7200
STA	3xxx		CME	7100
			HLT	7001

Refer to Chapter-5 for a description of the instructions.

Design the register set, the memory, and the instruction set. Use this machine for the assignments in this section.

1. Implement fetch sequence
2. Write an assembly program to simulate the addition of two numbers when one is stored in memory and another is entered by the user.
3. Write an assembly program to simulate addition of two numbers when both numbers are taken as inputs from user.
4. Write an assembly program to simulate subtraction of two numbers when one number is stored in memory and another is entered by the user.
5. Write an assembly program to simulate subtraction of two numbers when both numbers are taken as inputs from user
6. Write an assembly program to simulate the following logical operations on two user-entered numbers.

i.AND

ii.OR

iii.NOT

7. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution:

i. CLE

ii. CLA

iii. CMA

iv. CME

Note: Examination scheme and mode shall be as prescribed by the Examination Branch, University of Delhi, from time to time.