



SYLLABUS FOR DISCIPLINE SPECIFIC COURSE (DSC) WITH COMPUTER SCIENCE

Under Three Discipline Specific Course (FYUGP)
(To be implemented from Session 2024-25)

SEM. I & II

Proposed Syllabus for Discipline Specific Course (DSC) with Computer Science Programme							
Year	Semester	Paper Code	Paper	Credits	Periods/Week	Exam. Marks	Total Marks
1 st Year	I	COMSDSC101	Computer Fundamentals	3	3	60	80
		COMSDSC101T	Computer Fundamentals (Tutorial)	1	1	20	
		DSC	To be Selected from other Discipline Core	3	3	60	80
		DSCT/L	To be Selected from other Discipline Core	1	1/2	20	
		SEC1	To be chosen from a pool of subjects	2	2	60	60
		MIN1	Student has to choose only ONE discipline from the subjects provided by the University.	4	4		80
		VAC1	Understanding India	4	4		80
	II	COMSDSC202	Programming in C	3	3	60	80
		COMSDSC202L	Programming in C (Lab)	1	1	20	
		DSC	To be Selected from other Discipline Core	3	3	60	80
		DSCL/T	To be Selected from other Discipline Core	1	1/2	20	
		SEC2	To be chosen from a pool of subjects	2	2	60	60
		MIN2	Student will be provided the SAME discipline from the subjects selected previously as Minor.	4	4		80
		AEC1	One from MIL (Bengali/Nepali/Hindi/Urdu/Sanskrit/Alternative English)	4	4		80
		IDC1	Student has to choose only ONE discipline from the pool of subjects.	3	3		60
	IN1	Summer Internship The Colleges are expected to network with skill development centres, vocational training institutes for facilitating student internships. Online based internships programs are also permitted in case of Computer Science (Major) students. The students must submit a certificate of completion of the internship at the end of the semester.	2	-	-		

NOTE:

Tutorials should involve problem solving session/activity related to the subject taught.

1 st Year Semester-I			
Course-DSC Paper:	Paper Code- COMSDSC101 Computer Fundamentals	Credits-3	Lectures/Week-3

Prerequisite(s) and/or Note(s):

- (1) High school Physics.
- (2) Note(s): Syllabus changes yearly and may be modified during the term itself, depending on the circumstances. However, students will be evaluated only on the basis of topics covered in the course.

Course Objectives:

Knowledge acquired:

- (1) Introduce students to the fundamental concepts of system tools, their functionalities, and their role in computer systems.
- (2) Familiarize students with various peripheral devices commonly used in computer systems and their functionalities.
- (3) Develop students' proficiency in utilizing system tools to optimize computer performance, troubleshoot issues, and maintain system integrity.
- (4) Enable students to effectively manage peripheral devices, including printers, scanners, external storage devices, and input/output devices.

Skills gained:

- (1) Basic idea programming.
- (2) Troubleshooting hardware and software issues.
- (3) Efficient internet and file management.

Competency Developed:

- (1) Problem-solving in technical contexts.
 - (2) Adaptability to various software environments.
- Effective communication of technical concepts.

Syllabus Overview

Unit 1:	Basics of Computer	15 Lectures
Generation of Computers; Computer system : Basic Block Diagram, Super Mainframe, Mini & Personal Computer, Nomenclature, Software : Systems and Application; Hardware & Software; Algorithms : Definition, essential features;		
Unit 2:	Peripheral Devices	15 Lectures
Input and Output Devices – Punched Card, Keyboard, Mouse, Joystick, Trackball, Light Pen, Touch Screen, Magnetic Ink Character Recognition (MICR), Optical Character Recognition (OCR), Optical Mark Recognition (OMR), Display units, Printers- Impact and Non-Impact. Primary storage – RAM-SRAM, DRAM, ROM-PROM, EPROM, EEPROM, Secondary storage – Hard drive, Magnetic drive, Compact Disk, Cache memory, components of motherboard.		
Unit 3:	Programming Fundamentals	10 Lectures
Complexity: notation, time & space; Computability & correctness concepts; Structured programming concepts; Process of problem solving, Flowcharts and Pseudo codes.		

Unit 4:	Software and Data	5 Lectures
----------------	--------------------------	-------------------

Concept of software, Firmware, Types of Software: (System software, Application software, Utility software), Concept of Operating System, Loaders, Linkers, Debuggers, Translators: (Compilers, Assemblers, Interpreters), Data and Information, Types of data, Units of data measurement, Qualities of Information, Concept of database, Languages, Generation of Languages.

Suggested Readings

1. Sinha P.K., "Computer Fundamentals", 6 th Edition, BPB Publication, 2012.
2. Rajaraman,V, "Computer Fundamentals", 6 th Edition, PHI,2012.
3. Thareja R., "Fundamentals of Computers", Oxford University Press, 2014.
4. Stallings W., "Operating systems", 8th Edition, Pearson, 2014.

Course-DSC	Paper Code- COMSDSC101T	Credits-1	Tut./Week-1
Paper:	Computer Fundamentals (Tutorial)		

Computer Fundamentals Tutorial as assigned and advised by teacher(s).

**1st Year
Semester-II**

Course-DSC Paper:	Paper Code-COMSDSC202 Programming Fundamentals Using C	Credits-3	Lectures/Week-3
------------------------------	---	------------------	------------------------

Prerequisite(s) and/or Note(s):

- (1) High school mathematics.
- (2) Note(s): Syllabus changes yearly and may be modified during the term itself, depending on the circumstances. However, students will be evaluated only on the basis of topics covered in the course.

Course Objectives:

Knowledge acquired:

- (1) Knowledge about program development and implementation
- (2) Syntax of C programming language
- (3) Knowledge about how humans interact with computers through a language.

Skills gained:

- (1) Problem solving skills
- (2) Logical thinking to approach a problem
- (3) Building programs for different problems at hand.

Competency Developed:

- (1) Applying the skills learnt to model real world problems
- (2) Facility in solving real life problems by thinking logically and outside of box.
- (3) Ease of switching to any other programming language

Syllabus Overview

Unit 1:	Introduction to C programming	10 Lectures
----------------	--------------------------------------	--------------------

History of C, Overview of Procedural Programming, Introduction to Algorithm & Flowcharts. Using main() function Compiling and Executing Simple Programs in C. Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putchar etc), Formatted and Console I/O (printf(), scanf()) , Using Basic Header Files (stdio.h, conio.h, stdlib.h etc).

Unit 2:	Expression and Control Flow	10 Lectures
----------------	------------------------------------	--------------------

Simple Expressions in C (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Unit 3:	Functions	7 Lectures
----------------	------------------	-------------------

Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Return type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Functions with variable number of Arguments.

Unit 4: Arrays and Strings in C 10 Lectures

Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns),

Unit 5: User-defined Data Types and Pointers Basics 8 Lectures

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Pointers, integer, float and string pointers, working of pointers.

Suggested Readings

1. Rajaraman V. & Radhakrishnan, An Introduction To Digital Computer Design, PHI.
2. Malvino & Leach, Digital Principles & Applications, TMH
3. S. Salivahanan, S. Arivazhagan, Digital Circuits and Design, Oxford University Press

**Course-MAJOR Paper Code- COMSDSC202L Credits-1 Lab hours/Week-2
Paper: Programming Fundamentals Using C (Lab)**

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. WAP to perform input/output of all basic data types.
2. WAP to enter two numbers and find their sum.
3. WAP to reverse a number.
4. WAP to Swap Two Numbers (using and without using a third variable).
5. WAP to check whether a number is even or odd
6. WAP to compute the factors of a given number.
7. WAP to enter marks of five subjects and calculate total, average and percentage.
8. WAP to print the sum and product of digits of an integer.
9. WAP to check whether a character is vowel or consonant
10. WAP to find the largest among three numbers