MCA- C101 Programming Fundamentals and C Language				
	L	T	P	C
	4	0	0	4

Course objective:

1. To learn the fundamentals of computers, problem solving techniques by writing algorithms and procedures.
2. To learn the syntax and semantics of C programming language.

Course outcomes:

1. Ability to write algorithms for problems.
2. Ability to code a given logic in C language.

Fundamentals: Classification of Computers, Application of Computers, Basic organization of computer, Input and Output Devices, Binary Number System, Computer memory, Computer Software. Algorithm, Generation of Programming Languages.

Introduction to 'C' programming: Fundamentals, Structure of a 'C' program, Compilation and linking processes.

Expressions and Console I/O: Basic Data types, Identifier Names, Variables, Scope, Type qualifiers, Storage class specifiers, Constants, Operators, Reading and writing characters, Reading and writing strings, Formatted and console I/O, printf(), scanf(), Suppressing input.

Statements: True and False in C, Selection statements, Iteration statements, Jump statements, Expression statements, Block statements.

Arrays and Strings: Single dimensional array, Two-dimensional array, Strings, Array of strings, Multi-dimension array, Variable length arrays.

Pointers: Pointer variables, Pointer operators, Pointer expressions, Pointers and arrays, Multiple indirection, Pointer initialization, Pointers to arrays, Dynamically allocated arrays, Problems with pointers.

Functions: General form of a function, understanding scope of a function, Function arguments, Command line arguments, Return statement, Recursion, Function prototype, Pointers to functions.

Structures, Unions, Enumerations, and Typedef: Structures, Array of structures, passing structures to functions, Structure pointers, Arrays and structures within structures, Unions, Bit-fields, Enumerations, typedef.

File I/O: Streams and files, File system basics, fread() and fwrite(), fseek() and random access I/O, fprintf() and fscanf(), Standard streams.

Pre-processor and Comments: Pre-processor, #define, #error, #include, Conditional compilation directives, #undef, Single line and multiple line comments.

Recommended Books:

1. Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language, Prentice Hall
2. K.N. King, C Programming: A Modern Approach, W W Norton & Company, Inc

HEAD
Department of Computer Science
Gurukul Kangri Vishwavidyalaya
Haridwar (UK) - 249404