Department of Computer Science & Engineering
School of Engineering & Technology
HNB Garhwal University Srinagar Garhwal

Admission Notification for M. Tech. (Computer Science & Engineering) First Semester Batch 2017-2019

Applications are invited for admission in two years Full time M. Tech (Computer Science & Engineering) course for the session 2017-19 in the Department of Computer Science & Engineering latest by **07 September, 2017**. Admissions shall be given strictly on the basis of result of entrance examination (if the number of candidates is greater than the prescribed no. of seats) to be held on **09 September, 2017**. The written test will be objective type of one hour duration. The candidates may visit university website **http://hnbguentrance.in/admission/** for on-line form filling.

Eligibility Criteria

Course Name	**Number of Seats**	**Eligibility Criteria**
M.Tech. (Computer Science & Engineering)	10	Minimum 60% or equivalent grade in 4 year B.E. /B. Tech in Electronics & Communication Engg /Electronics Engg. /Electrical Engg. /Instrumentation Engg. / Instrumentation & Control Engg/Computer Science &Engg/ Information Technology/ or M.Sc. (Computer Science)/MCA

Head
Department of Computer Science & Engineering

Syllabus for M.Tech. (Computer Science & Engineering) Entrance Examination-2017

Computer Organization and Architecture

Representation of information and Basic Building Blocks: Number System: Binary, Octal, Hexadecimal and their conversion, Character Codes: BCD, ASCII, EBCDIC. Digital Codes: Gray Code, XS-3 Code.

Logic circuits : Basic Logic Functions, Synthesis of Logic Functions Using AND, OR and NOT Gates, Minimization of Logic Expression, Synthesis with NAND and NOR Gates, Implementation of Logic Gates, Flip-Flops, Registers and Shift Registers, Counters, Decoders, Multiplexers, Programmable Logic Devices , Sequential Circuits.

Basic Structure of Computer Hardware and Software: Functional units, Basic operational concepts, Bus structures, Software, Performance, Distributed Computing.

Addressing Methods : Basic Concepts, Memory Locations, Main Memory Operations, Addressing Modes, Basic I/O operations, Stacks and Queues, Subroutines.

Processing Unit : Some Fundamental Concepts, Execution of a Complete Instruction, Hardwired Control, Performance Considerations, Micro Programmed Control, Signed Addition and Subtraction, Arithmetic and Branching Conditions, Multiplication of Positive Numbers, Signed-Operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations.

Input-output Organization: Accessing I/O Devices, Interrupts, Direct Memory Access, I/O Hardware, Standard I/O Interfaces. Memory: Semiconductor RAM memories, Read-Only Memories, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements.

Introduction to Computer Peripherals: I/O Devices, On-Line Storage.

C Programming

Basic Programming Concepts: Introduction to the basic ideas of problem solving and programming using principles of top-down modular design, Flowcharts, Abstraction Mechanisms, Stepwise Refinement.

Introduction to Programming Language C: Data Types, Instruction and its Types, Storage Classes, C character set, Identifiers and keywords, Data types, Declarations, Expressions, statements and symbolic constants, #include, define, if def. Preparing and running a complete C program.

Operators and expressions: Arithmetic, unary, logical, bit-wise, assignment and conditional operators, Library functions, Control statements: while, do-while, for statements, nested loops. If-else, switch, break, continue and go to statements, comma operator.

Functions: Defining and accessing: passing arguments, Function prototypes, Recursion, Use of library functions. Storage classes: automatic, external and static variables. Arrays: Defining and processing, passing to a function, Multi-dimensional arrays. Strings: Operations on strings. Pointers: Declarations, Passing to a function, Operations on pointers, Pointers and arrays, Arrays of pointers. Structures: Defining and processing, passing to a function, Unions. Data files: Open, close, create, process, Unformatted data files.

Data and File Structure

Introduction: Basic Terminology, Elementary Data Organization, Data Structure operations, Algorithm Complexity and Time-Space trade-off. Arrays: Array Definition, Representation and Analysis, Single and Multidimensional Arrays, address calculation, application of arrays, Character String in C, Character string operation, Array as Parameters, Ordered list, Sparse Matrices, and Vector. Stacks: Array Representation and Implementation of stack, Operations and Stacks: Push and POP, Array Representation of Stack, Linked Representation of stack, Operations Associated with Stacks, Application of stack, Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack. Recursion: Recursive definition and processes, recursion in C, example of recursion, Tower of Hanoi Problem.

Queues: Array and linked representation and implementation of queues, Operations on Queue; Create, Add, Delete, Full and Empty, Circular queue, Dequeue, and Priority Queue. Link List: Representation and implementation of Singly linked lists, Two-way Header List, Traversing and Searching of Linked List, Overflow and Underflow,

Insertion and deletion to from Linked Lists, Insertion and deletion Algorithms, Doubly linked list, Linked List of Array, Polynomial representation and addition, Generalized linked list, Garbage Collection and Compaction.

Trees: Basic terminology, Binary Tree, Binary tree representation algebraic Expressions, Complete Binary Tree, Extended Binary Tree, Array and Linked Representation of Binary trees, Traversing Binary trees, Threaded Binary trees. Traversing Threaded Binary tree, Huffman algorithm. Searching and Hashing: Sequential search, comparison and analysis, Hash Table, Hash Function, Collection Resolution Strategies, Hash Table Implementation.

Sorting: Insertion Sort, Bubble sorting, Quick Sort, Two way Merge Sort, Heap Sort, Sorting on Different Keys, Practical Consideration for Internal Sorting. Binary Search Trees, AVL Tree, B-trees.

File Structures: Physical Storage Media File Organization, Organization of records into Blocks, Sequential Files, Indexing and Hashing, Primary indices, and Secondary indices.

Theory of Computation

Introduction to the Theory of computation and Finite Automata: Mathematical preliminaries and Notation, three basic concepts, applications, deterministic Finite Acceptors, Nondeterministic finite acceptors, equivalence of Deterministic and Nondeterministic finite acceptors, reduction of the Number of states in Finite Automata.

Regular Languages, regular grammars and Properties of Regular Languages: regular expressions, connection between regular expressions and regular languages, regular grammars, closure properties of regular languages, elementary questions about regular languages, identifying language.

Contest-free languages and simplification of context-free grammars and normal forms: context-free grammars, parsing and ambiguity, context-free grammars and programming languages, methods of transforming grammars, two important normal forms.

Pushdown automata and properties of context-free languages: Non-deterministic pushdown automata, pushdown automata and context-free language, deterministic pushdown automata and deterministic context-free languages, two pumping lemmas, closure properties and decision algorithms for context-free language.

Turning machines and other models of turning machines: the standard turning machine, combining turning machines for complicated tasks, Turing's thesis, minor variation on the turning machine, combining turning machines, a universal turning machine.

Analysis and Design of Algorithms

Introduction: Algorithms, Analysis of Algorithms, Design of Algorithms, and Complexity of Algorithms, Asymptotic Notations, Growth of function, Recurrences. Sorting in polynomial Time: Insertion sort, Merge sort, Heap sort, and Quick sort Sorting in Linear Time: Counting sort, Radix Sort, Bucket Sort Medians and order statistics.

Advanced Data Structure: Red Black Trees, Splay Trees, Augmenting Data Structure Binomial Heap, B-Tree, Fibonacci Heap, and Data structure for Disjoint Sets. Union-find Algorithm, Dictionaries and priority Queues, mergeable heaps, concatenable queues.

Advanced Design and Analysis Techniques: Dynamic Programming, Greedy Algorithm, Backtracking, Branch-and-Bound, Amortized Analysis.

Graph Algorithms: Elementary Graph Algorithms, Breadth First search, Depth First search, Minimum Spanning Tree, Kruskal's Algorithms, Prim's Algorithms, Single Source Shortest Path, All pair Shortest Path, Maximum flow and Traveling Salesman Problem.

Randomized Algorithms, String Matching, NP-Hard and NP-Completeness Approximation Algorithms, Sorting Network, Matrix Operations, Polynomials and the FFT, Number Theoretic Algorithms.

Theory of Computation

Introduction to the Theory of computation and Finite Automata: Mathematical preliminaries and Notation, three basic concepts, applications, deterministic Finite Acceptors, Nondeterministic finite acceptors, equivalence of Deterministic and Nondeterministic finite acceptors, reduction of the Number of states in Finite Automata.

Regular Languages, regular grammars and Properties of Regular Languages: regular expressions, connection between regular expressions and regular languages, regular grammars, closure properties of regular languages, elementary questions about regular languages, identifying language.

Contest-free languages and simplification of context-free grammars and normal forms: context-free grammars, parsing and ambiguity, context-free grammars and programming languages, methods of transforming grammars, two important normal forms.

Pushdown automata and properties of context-free languages: Non-deterministic pushdown automata, pushdown automata and context-free language, deterministic pushdown automata and deterministic context-free languages, two pumping lemmas, closure properties and decision algorithms for context-free language.

Turning machines and other models of turning machines: the standard turning machine, combining turning machines for complicated tasks, Turing's thesis, minor variation on the turning machine, combining turning machines, a universal turning machine.

Software Engineering

Introduction: Introduction to software engineering, Importance of software, evolving role of software, Software Characteristics, Software Components, Software Applications, Software Crisis, Software engineering problems, Software Development Life Cycle, Software Process.

Software Requirement Specification: Analysis, Principles, Water Fall Model, The Incremental Model, Prototyping, Spiral Model, Role of management in software development, Role of matrices and Measurement, Problem Analysis, Requirement specification, Monitoring and Control.

Software-Design: Design principles, problem partitioning, abstraction, top down and bottom up-design, Structured approach functional versus object oriented approach, design specifications and verification, Monitoring and control, Cohesiveness, coupling, Fourth generation techniques, Functional independence, Software Architecture, Transaction and Transaction and Transform Mapping, Component level Design, Fourth Generation Techniques.

Coding: Top-Down and Bottom Up programming, structured programming, information hiding, programming style and internal documentation.

Testing principles, Levels of testing, functional testing, structural testing, test plane, test case specification, reliability assessment, software testing strategies, Verification and validation, Unit testing, Integration Testing, Alpha and Beta testing, system testing and debugging.

Compiler Designing

Compiler Structure: Compilers and Translators, Various Phases of Compiler, Pass Structure of Compiler, Bootstrapping of Compiler.

Programming Language: High level languages, lexical and syntactic structure of a language, Data elements, Data Structure, Operations, Assignments, Program unit, Data Environments, Parameter Transmission. Lexical Analysis: The role of Lexical Analyzer, A Simple approach to the design of Lexical Analyzer, Regular Expressions, Transition Diagrams, Finite state Machines, Implementation of Lexical Analyzer, Lexical Analyzer Generator: LEX, Capabilities of Lexical Analyzer.

The Syntactic Specification of Programming Languages: CFG, Derivation and Parse tree, Ambiguity, Capabilities of EFG.

Basic Parsing Techniques: Top-Down parsers with backtracking, Recursive descent Parsers, Predictive Parser, Bottom-up Parsers, Shift-Reduce Parsing, Operator Precedence Parsers, LR parsers (SLR, Canonical LR, LALR) Syntax Analyzer Generator: YACC

Intermediate Code Generation: Different Intermediate forms: Three address code, Quadruples and Triples, Syntax Directed Translation mechanism and attributed definition. Translation of Declaration, Assignment, Control flow, Boolean expression, Array References in arithmetic expressions, procedure calls, case statements, postfix translation.

Run Time Memory Management: Static and Dynamic storage allocation, stack based memory allocation schemes, Symbol Table management.

Error Detection and Recovery: Lexical phase errors. Syntactic phase errors, Semantic errors.

Code Optimization and Code Generation: Local optimization, Peephole optimization, Basic blocks and flow Graphs, DAG, Data flow analyzer, Machine Model, Order of evaluation, Register allocation and code selection.

Operating Systems

Introduction, What is an Operating System, Simple Batch Systems, Multiprogrammed Batches systems, Time-Sharing Systems, Personal-computer systems, Parallel systems, Distributed Systems, Real-Time Systems.

Memory Management: Background, Logical versus Physical Address space, swapping, Contiguous allocation, Paging, Segmentation, Segmentation with Paging Virtual Memory: Demand Paging, Page Replacement, Page-replacement Algorithms, Performance of Demand Paging, Allocation of Frames, Thrashing, Other Considerations, Demand Segmentation.

Processes: Process Concept, Process Scheduling, Operation on Processes, Cooperating Processes, Inter-process Communication

CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple- Processor Scheduling, Real-Time Scheduling, Algorithm Evaluation.

Process Synchronization: Background, Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors, Synchronization in Solaris 2, Atomic Transactions.

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Combined Approach to Deadlock Handling.

Device Management: Techniques for Device Management, Dedicated Devices, Shared Devices, Virtual Devices; Device Characteristics-Hardware Consideration, Input or Output Devices, Storage Devices, Channels and Control Units, Independent Device Operation, Buffering, Multiple Paths, Block Multiplexing, Device Allocation Consideration, Secondary-Storage Structure: Disk Structure, Disk Scheduling, Disk Management, Swap-Space Management, Disk Reliability, Stable-Storage Implementation.

Database Management System

Concept & Overview of DBMS, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS.

Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended E-R features.

Relational Model: Structure of relational Databases, Relational Algebra, Relational Calculus, Extended Relational Algebra Operations, Views, Modifications Of the Database.

SQL and Integrity Constraints: Concept of DDL, DML, DCL. Basic Structure, Set operations, Aggregate Functions, Null Values, Domain Constraints,

Referential Integrity Constraints, assertions, views, Nested Subqueries, Database security application development using SQL, Stored procedures and triggers.

Relational Database Design: Functional Dependency, Different anomalies in designing a Database, Normalization using functional dependencies,

Decomposition, Boyce-Codd Normal Form, 3NF, Normalization using multi-valued dependencies, 4NF, 5NF

Internals of RDBMS: Physical data structures, Query optimization : join algorithm, statistics and cost bas optimization. Transaction processing, Concurrency control and Recovery Management: transaction model properties, state serializability, lock base protocols, two phase locking.

File Organization & Index Structures: File & Record Concept, Placing file records on Disk, Fixed and Variable sized Records, Types of Single-Level Index (primary, secondary, clustering), Multilevel Indexes, Dynamic Multilevel Indexes using B tree and B+ tree .

Computer Networks

Introduction and The Physical Layer: Uses of Computer Networks, Network Hardware, Network Software, Topology, Network types, Reference Model (OSI, TCP/IP Overview), The Physical Layer, Theoretical Basis for Data Communication, Guided Transmission Media, Wireless Transmission, Communication Satellites, Digital and Analog Signal, FSK, PSK, modulation techniques, Switching techniques.

The Data Link Layer: Data Link Layer Design Issues, Error Detection and Correlation, Flow Control Protocols, Stop-and-wait Flow Control, Sliding – Window Flow Control, Error Control, Stop-and-wait ARQ, Go-back-N, Selective-repeat, Example of Data Link Protocols- HDLC.

The Medium Access Control Sub Layer: The Channel Allocation Problem, Multiple Access Protocols, Ethernet, wireless LANs, Blue Tooth, Data Link Layer Switching.

The Network Layer: Network Layer Design Issues, Routing, Quality of Service, Internetworking,

The Transport Layer : The Transport layer Services, Elements of Transport Protocols, A Simple Transport Protocol, The Internet Transport Protocols; UDP, TCP, Performance Issues, Congestion control.

Application Layer: Network Security, DES, RSA algorithms, Domain Name System, Simple Network Management Protocol, Electronic mail, File Transfer Protocol, Hyper Text Transfer Protocol, Cryptography and compression Techniques.

Unix & Shell Programming

Basic Unix Commands: The Unix editors and vi; Redirection, Piping, Tees and filters; The Unix Utilities grep, sed, etc.

Overview of Unix Architecture: The kernel and the Shell; Processes and Time Sharing files and Directories; Peripheral Device as files.

Introduction the Shell Scripts: The Bourne and C-shells; Shell variables, scripts meta-characters and environment; if and case statements; for, while and until loops.

System calls and the 'C' library: Discussion of the Unix system calls and 'C' library functions the standard I/O Package; file handling; math library; command line parameters etc. The Unix 'C' interface; 'C' files and Graphics.

Introduction of systems Administration under Unix: The system Manger OLE and functions.

Bourne Shell: Shell meta characteristics, shell variable, scripts, facilities, commands and environments, shell archive, idea about restricted shell, ROLC program.

Korn Shell: Shell variables and scripts, built in EDITOR, built in integer arithmetic, string manipulation capabilities, Command Aliasing, Array Job control.