V.B.S. PURVANCHAL UNIVERSITY, JAUNPUR

Department of Computer Science & Engineering

Evaluation Scheme & Syllabus

For

B.Tech. 2nd Year (III & IV Semester)

(Computer Science and Engineering)

(Effective from the Session 2020-21)

V.B.S. PURVANCHAL UNIVERSITY JAUNPUR

B.TECH (COMPUTER SCIENCE AND ENGINEERING)

SEMESTER- III

Sl. No.	Subject Codes	Subject	Periods			Evaluation Scheme				End Semester		Total	Credit
			L	T	P	CT	TA	Total	PS	TE	PE		
1	KOE031-38/ KAS302	Engineering Science Course/Maths IV	3	1	0	30	20	50		100		150	4
2	KAS301/ KVE 301	Technical Communication/Universal Human values	2 3	1 0	0 0	30	20	50		100		150	3
3	KCS301	Data Structure	3	1	0	30	20	50		100		150	4
4	KCS302	Computer Organization and Architecture	3	1	0	30	20	50		100		150	4
5	KCS303	Discrete Structures & Theory of Logic	3	0	0	30	20	50		100		150	3
6	KCS351	Data Structures Using C Lab	0	0	2				25		25	50	1
7	KCS352	Computer Organization Lab	0	0	2				25		25	50	1
8	KCS353	Discrete Structure & Logic Lab	0	0	2				25		25	50	1
9	KCS354	Mini Project or Internship Assessment*	0	0	2			50				50	1
10	KNC301/ KNC302	Computer System Security/Python Programming	2	0	0	15	10	25		50			0
		Total										**950**	**22**

*The Mini Project or internship (3-4 weeks) conducted during summer break after II semester and will be assessed during III semester.

SEMESTER- IV													
Sl. No.	**Subject Codes**	**Subject**	**Periods**			**Evaluation Scheme**				**End Semester**		**Total**	**Credit**
			L	**T**	**P**	**CT**	**TA**	**Total**	**PS**	**TE**	**PE**		
1	KAS402/ KOE041-48	Maths IV/Engg. Science Course	3	1	0	30	20	50		100		150	4
2	KVE401/ KAS301	Universal Human Values/ Technical Communication	3 2	0 1	0 0	30	20	50		100		150	3
3	KCS401	Operating Systems	3	0	0	30	20	50		100		150	3
4	KCS402	Theory of Automata and Formal Languages	3	1	0	30	20	50		100		150	4
5	KCS403	Microprocessor	3	1	0	30	20	50		100		150	4
6	KCS451	Operating Systems Lab	0	0	2				25		25	50	1
7	KCS452	Microprocessor Lab	0	0	2				25		25	50	1
8	KCS453	Python Language Programming Lab	0	0	2				25		25	50	1
9	KNC402/ KNC401	Python Programming/Computer System Security	2	0	0	15	10	25		50			0
		Total										**900**	**21**

B.TECH. (COMPUTER SCIENCE AND ENGINEERING)
THIRD SEMESTER (DETAILED SYLLABUS)

DATA STRUCTURE (KCS301)		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course , the student will be able to understand		
CO 1	Describe how arrays, linked lists, stacks, queues, trees, and graphs are represented in memory, used by the algorithms and their common applications.	**K_1, K_2**
CO 2	Discuss the computational efficiency of the sorting and searching algorithms.	**K_2**
CO 3	Implementation of Trees and Graphs and perform various operations on these data structure.	**K_3**
CO 4	Understanding the concept of recursion, application of recursion and its implementation and removal of recursion.	**K_4**
CO 5	Identify the alternative implementations of data structures with respect to its performance to solve a real world problem.	**K_5, K_6**
DETAILED SYLLABUS		**3-1-0**

Unit	Topic	Proposed Lecture
I	**Introduction:** Basic Terminology, Elementary Data Organization, Built in Data Types in C. Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations: Big Oh, Big Theta and Big Omega, Time-Space trade-off. Abstract Data Types (ADT) **Arrays**: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D,2-D,3-D and n-D Array Application of arrays, Sparse Matrices and their representations. **Linked lists:** Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition Subtraction & Multiplications of Single variable & Two variables Polynomial.	08
II	**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Iteration and Recursion- Principles of recursion, Tail recursion, Removal of recursion Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. Tradeoffs between iteration and recursion. **Queues:** Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.	08
III	**Searching:** Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing. **Sorting:** Insertion Sort, Selection, Bubble Sort, Quick Sort, Merge Sort, Heap Sort and Radix Sort.	08
IV	**Graphs:** Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List, Adjacency. Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Transitive Closure and Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm.	08
V	**Trees:** Basic terminology used with Tree, Binary Trees, Binary Tree Representation: Array Representation and Pointer(Linked List) Representation, Binary Search Tree, Strictly Binary Tree ,Complete Binary Tree . A Extended Binary Trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Constructing Binary Tree from given Tree Traversal, Operation of Insertation , Deletion, Searching & Modification of data in Binary Search . Threaded Binary trees, Traversing Threaded Binary trees. Huffman coding using Binary Tree. Concept & Basic Operations for AVL Tree , B Tree & Binary Heaps	08

Text books:

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, "Data Structures Using C and C++", PHI Learning Private Limited, Delhi India
2. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.
3. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.
4. Thareja, "Data Structure Using C" Oxford Higher Education.
5. AK Sharma, "Data Structure Using C", Pearson Education India.
6. Rajesh K. Shukla, "Data Structure Using C and C++" Wiley Dreamtech Publication.
7. Michael T. Goodrich, Roberto Tamassia, David M. Mount "Data Structures and Algorithms in C++", Wiley India.
8. P. S. Deshpandey, "C and Data structure", Wiley Dreamtech Publication.
9. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education.
10. Berztiss, AT: Data structures, Theory and Practice, Academic Press.
11. Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with applications", McGraw Hill.
12. Adam Drozdek "Data Structures and Algorithm in Java", Cengage Learning

Computer Organization and Architecture (KCS302)		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course , the student will be able to understand		
CO 1	Study of the basic structure and operation of a digital computer system.	K_1, K_2
CO 2	Analysis of the design of arithmetic & logic unit and understanding of the fixed point and floating-point arithmetic operations.	K_2, K_4
CO 3	Implementation of control unit techniques and the concept of Pipelining	K_3
CO 4	Understanding the hierarchical memory system, cache memories and virtual memory	K_2
CO 5	Understanding the different ways of communicating with I/O devices and standard I/O interfaces	K_2, K_4

DETAILED SYLLABUS		**3-1-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction**: Functional units of digital system and their interconnections, buses, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer. Processor organization, general registers organization, stack organization and addressing modes.	**08**
II	**Arithmetic and logic unit:** Look ahead carries adders. Multiplication: Signed operand multiplication, Booths algorithm and array multiplier. Division and logic operations. Floating point arithmetic operation, Arithmetic & logic unit design. IEEE Standard for Floating Point Numbers	**08**
III	**Control Unit:** Instruction types, formats, instruction cycles and sub cycles (fetch and execute etc), micro operations, execution of a complete instruction. Program Control, Reduced Instruction Set Computer, Pipelining. Hardwire and micro programmed control: micro programme sequencing, concept of horizontal and vertical microprogramming.	**08**
IV	**Memory:** Basic concept and hierarchy, semiconductor RAM memories, 2D & 2 1/2D memory organization. ROM memories. Cache memories: concept and design issues & performance, address mapping and replacement Auxiliary memories: magnetic disk, magnetic tape and optical disks Virtual memory: concept implementation.	**08**
V	**Input / Output**: Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access., I/O channels and processors. Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.	**08**

Text books:

1. Computer System Architecture - M. Mano
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky Computer Organization, McGraw-Hill, Fifth Edition, Reprint 2012
3. John P. Hayes, Computer Architecture and Organization, Tata McGraw Hill, Third Edition, 1998. Reference books
4. William Stallings, Computer Organization and Architecture-Designing for Performance, Pearson Education, Seventh edition, 2006.
5. Behrooz Parahami, "Computer Architecture", Oxford University Press, Eighth Impression, 2011.
6. David A. Patterson and John L. Hennessy, "Computer Architecture-A Quantitative Approach", Elsevier, a division of reed India Private Limited, Fifth edition, 2012
7. Structured Computer Organization, Tannenbaum(PHI)

Data Structure using C Lab (KCS351)

Write C Programs to illustrate the concept of the following:

1. Sorting Algorithms-Non-Recursive.
2. Sorting Algorithms-Recursive.
3. Searching Algorithm.
4. Implementation of Stack using Array.
5. Implementation of Queue using Array.
6. Implementation of Circular Queue using Array.
7. Implementation of Stack using Linked List.
8. Implementation of Queue using Linked List.
9. Implementation of Circular Queue using Linked List.
10. Implementation of Tree Structures, Binary Tree, Tree Traversal, Binary Search Tree, Insertion and Deletion in BST.
11. Graph Implementation, BFS, DFS, Minimum cost spanning tree, shortest path algorithm.

Discrete Structure & Logic Lab (KCS353)

Programming Language/Tool Used: C and Mapple

1. Write a program in C to create two sets and perform the Union operation on sets.
2. Write a program in C to create two sets and perform the Intersectison operation on sets.
3. Write a program in C to create two sets and perform the Difference operation on sets.
4. Write a program in C to create two sets and perform the Symmetric Difference operation.
5. Write a program in C to perform the Power Set operation on a set.
6. Write a program in C to Display the Boolean Truth Table for AND, OR , NOT .
7. Write a C Program to find Cartesian Product of two sets
8. Write a program in C for minimum cost spanning tree.
9. Write a program in C for finding shortest path in a Graph.
10. Write a program in C for finding the inorder, preorder,postorder.
11. Write a program in C for Binary Tree.
12. Write a program in C for Binary Search Tree.

B.TECH. (COMPUTER SCIENCE AND ENGINEERING)

FOURTH SEMESTER (DETAILED SYLLABUS)

Operating systems (KCS401)		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course , the student will be able to understand		
CO 1	Understand the structure and functions of OS	K_1, K_2
CO 2	Learn about Processes, Threads and Scheduling algorithms.	K_1, K_2
CO 3	Understand the principles of concurrency and Deadlocks	K_2
CO 4	Learn various memory management scheme	K_2
CO 5	Study I/O management and File systems.	K_2, K_4

DETAILED SYLLABUS		**3-0-0**
Unit	**Topic**	**Proposed Lecture**
I	**Introduction** : Operating system and functions, Classification of Operating systems- Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multiprocess Systems, Multithreaded Systems, Operating System Structure- Layered structure, System Components, Operating System services, Reentrant Kernels, Monolithic and Microkernel Systems.	**08**
II	**Concurrent Processes**: Process Concept, Principle of Concurrency, Producer / Consumer Problem, Mutual Exclusion, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, Test and Set operation; Classical Problem in Concurrency- Dining Philosopher Problem, Sleeping Barber Problem; Inter Process Communication models and Schemes, Process generation.	**08**
III	**CPU Scheduling:** Scheduling Concepts, Performance Criteria, Process States, Process Transition Diagram, Schedulers, Process Control Block (PCB), Process address space, Process identification information, Threads and their management, Scheduling Algorithms, Multiprocessor Scheduling. Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.	**08**
IV	**Memory Management:** Basic bare machine, Resident monitor, Multiprogramming with fixed partitions, Multiprogramming with variable partitions, Protection schemes, Paging, Segmentation, Paged segmentation, Virtual memory concepts, Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing, Cache memory organization, Locality of reference.	**08**
V	**I/O Management and Disk Scheduling**: I/O devices, and I/O subsystems, I/O buffering, Disk storage and disk scheduling, RAID. File System: File concept, File organization and access mechanism, File directories, and File sharing, File system implementation issues, File system protection and security.	**08**

Text books:

1. Silberschatz, Galvin and Gagne, "Operating Systems Concepts", Wiley
2. Sibsankar Halder and Alex A Aravind, "Operating Systems", Pearson Education
3. Harvey M Dietel, " An Introduction to Operating System", Pearson Education
4. D M Dhamdhere, "Operating Systems : A Concept based Approach", 2nd Edition,
5. TMH 5. William Stallings, "Operating Systems: Internals and Design Principles ", 6th Edition, Pearson Education

Theory of Automata and Formal Languages (KCS402)		
Course Outcome (CO)		**Bloom's Knowledge Level (KL)**
At the end of course , the student will be able to understand		
CO 1	Analyse and design finite automata, pushdown automata, Turing machines, formal languages, and grammars	K_4, K_6
CO 2	Analyse and design, Turing machines, formal languages, and grammars	K_4, K_6
CO 3	Demonstrate the understanding of key notions, such as algorithm, computability, decidability, and complexity through problem solving	K_1, K_5
CO 4	Prove the basic results of the Theory of Computation.	K_2, K_3
CO 5	State and explain the relevance of the Church-Turing thesis.	K_1, K_5

DETAILED SYLLABUS		**3-1-0**
Unit	**Topic**	**Proposed Lecture**
I	**Basic Concepts and Automata Theory:** Introduction to Theory of Computation- Automata, Computability and Complexity, Alphabet, Symbol, String, Formal Languages, Deterministic Finite Automaton (DFA)- Definition, Representation, Acceptability of a String and Language, Non Deterministic Finite Automaton (NFA), Equivalence of DFA and NFA, NFA with ε-Transition, Equivalence of NFA's with and without ε-Transition, Finite Automata with output- Moore Machine, Mealy Machine, Equivalence of Moore and Mealy Machine, Minimization of Finite Automata, Myhill-Nerode Theorem, Simulation of DFA and NFA	**08**
II	**Regular Expressions and Languages:** Regular Expressions, Transition Graph, Kleen's Theorem, Finite Automata and Regular Expression- Arden's theorem, Algebraic Method Using Arden's Theorem, Regular and Non-Regular Languages- Closure properties of Regular Languages, Pigeonhole Principle, Pumping Lemma, Application of Pumping Lemma, Decidability- Decision properties, Finite Automata and Regular Languages, Regular Languages and Computers, Simulation of Transition Graph and Regular language.	**08**
III	**Regular and Non-Regular Grammars**: Context Free Grammar(CFG)-Definition, Derivations, Languages, Derivation Trees and Ambiguity, Regular Grammars-Right Linear and Left Linear grammars, Conversion of FA into CFG and Regular grammar into FA, Simplification of CFG, Normal Forms- Chomsky Normal Form(CNF), Greibach Normal Form (GNF), Chomsky Hierarchy, Programming problems based on the properties of CFGs.	**08**
IV	**Push Down Automata and Properties of Context Free Languages**: Nondeterministic Pushdown Automata (NPDA)- Definition, Moves, A Language Accepted by NPDA, Deterministic Pushdown Automata(DPDA) and Deterministic Context free Languages(DCFL), Pushdown Automata for Context Free Languages, Context Free grammars for Pushdown Automata, Two stack Pushdown Automata, Pumping Lemma for CFL, Closure properties of CFL, Decision Problems of CFL, Programming problems based on the properties of CFLs.	**08**
V	**Turing Machines and Recursive Function Theory** : Basic Turing Machine Model, Representation of Turing Machines, Language Acceptability of Turing Machines, Techniques for Turing Machine Construction, Modifications of Turing Machine, Turing Machine as Computer of Integer Functions, Universal Turing machine, Linear Bounded Automata, Church's Thesis, Recursive and Recursively Enumerable language, Halting Problem, Post's Correspondance Problem, Introduction to Recursive Function Theory.	**08**

Text books:

1. Introduction to Automata theory, Languages and Computation, J.E.Hopcraft, R.Motwani, and Ullman. 2nd edition, Pearson Education Asia
2. Introduction to languages and the theory of computation, J Martin, 3rd Edition, Tata McGraw Hill
3. Elements and Theory of Computation, C Papadimitrou and C. L. Lewis, PHI
4. Mathematical Foundation of Computer Science, Y.N.Singh, New Age Internationa

Operating Systems Lab (KCS451)

1. Study of hardware and software requirements of different operating systems (UNIX,LINUX,WINDOWS XP, WINDOWS7/8
2. Execute various UNIX system calls for
    i. Process management
    ii. File management
    iii. Input/output Systems calls
3. Implement CPU Scheduling Policies:
    i. SJF
    ii. Priority
    iii. FCFS
    iv. Multi-level Queue
4. Implement file storage allocation technique:
    i. Contiguous(using array)
    ii. Linked –list(using linked-list)
    iii. Indirect allocation (indexing)
5. Implementation of contiguous allocation techniques:
    i. Worst-Fit
    ii. Best- Fit
    iii. First- Fit
6. Calculation of external and internal fragmentation
    i. Free space list of blocks from system
    ii. List process file from the system
7. Implementation of compaction for the continually changing memory layout and calculate total movement of data
8. Implementation of resource allocation graph RAG)
9. Implementation of Banker‟s algorithm
10. Conversion of resource allocation graph (RAG) to wait for graph (WFG) for each type of method used for storing graph.
11. Implement the solution for Bounded Buffer (producer-consumer)problem using inter process communication techniques-Semaphores
12. Implement the solutions for Readers-Writers problem using inter process communication technique -Semaphore

Python Language Programming Lab (KCS453)

1. To write a python program that takes in command line arguments as input and print the number of arguments.
2. To write a python program to perform Matrix Multiplication.
3. To write a python program to compute the GCD of two numbers.
4. To write a python program to find the most frequent words in a text file.
5. To write a python program find the square root of a number (Newton’s method).
6. To write a python program exponentiation (power of a number).
7. To write a python program find the maximum of a list of numbers.
8. To write a python program linear search.
9. To write a python program Binary search.
10. To write a python program selection sort.
11. To write a python program Insertion sort.
12. To write a python program merge sort.
13. To write a python program first n prime numbers.
14. To write a python program simulate bouncing ball in Pygame.