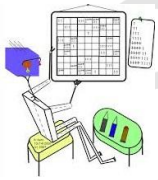
Continuous Evaluation

A set of knowledge based assessments that cover technical aspects of student's job readiness.

A **Summative Assessment** which is devised as per engineering semester frame work to examine student's knowledge level in **Industry Relevant** courses across major engineering branches which include **Civil, Mechanical, Electrical, Electronics and Computer Science Engineering**.



Assessment Frame work for Continuous Evaluation-Computer Science

1st/2nd Semester	• Basics of Computer Programming
3rd Semester	• Computer Architecture • Digital circuits & electronics • OOPS using C++/Java
4th Semester	• Computer Networks- I • Data structures • Operating Systems
5th/ 6th Semester	• Computer Networks-II • Design of Algorithms

Detailed Syllabus for Continuous Evaluation- Computer Science Engineering

- **Basics of Computer Programming -** Introduction, C programming basics, Arrays and strings, Functions and pointers, Structures and unions
- **Computer Architecture -** Computer organization and architecture-functional units – basic operational concepts, basic processing unit, pipelining, memory system, i/o organization

- **Digital Circuits - N**umber systems & boolean algebra, implementation of combinational logic design, design of synchronous sequential circuits, asynchronous sequential logic, hardware description logic

- **OOPS using C++/Java -** Introduction, C++ Programming basics, Functions, Object and Classes, Arrays and string arrays fundamentals. Arrays as class Member Data, Operator overloading, Inheritance, Pointer, Virtual Function, Templates and Exceptions, Java: The history & evaluation of java, Data types, Variables and arrays, operators, classes, swings, beans, AWT, Exceptional handling, Inheritance, Overloading, Packages and interface, Multithreading, Applets, Servelts, generics, Networking, JDBC

- **Computer Networks- I -** Overview of data communication and Networking, Reference models, Physical Layer and Media, Data Link Layer, Network Layer, Transport Layer, Application Layer, Security

- **Data Structures –** Introduction, Arrays, Linked lists, Stacks, Queue, Trees : Binary Trees, AVL tree, Minimum Cost Spanning Trees: Prims and Kruskal algorithm, Graphs, Graph Traversal : Depth First Search and Breadth First Search Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm, Searching, Sorting algorithms, Search Trees: Binary Search Trees(BST), Insertion and Deletion in BST, Complexity of Search Algorithm, AVL trees, Introduction to m-way Search Trees, B Trees & B+ Trees . Hashing: Hash Function

- **Operating Systems -** Introduction, Concurrent Processes, Mutual Exclusion, Critical Section Problems, CPU Scheduling, Threads and their management, Deadlock, Memory Management, I/O Management and Disk Scheduling

- **Computer Networks-II -** Overview of data communication and Networking, Reference models, Physical Layer and Media, Data Link Layer, Network Layer, Transport Layer, Application Layer Security

- **Design of Algorithms -** Introduction: Algorithms, Analyzing algorithms, Complexity of algorithms, Advanced Data Structures: Red-Black trees, B – trees, Binomial Heaps, Fibonacci Heaps. Divide and Conquer with examples such as Sorting, Matrix Multiplication and Searching. Greedy methods, Dynamic programming with examples such as Kanpsack, All pair shortest paths – Warshal's and Floyd's algorithms, Resource allocation problem. Backtracking, Branch and Bound with examples such as Travelling Salesman Problem, Graph Coloring, n-Queen Problem, Hamiltonian Cycles and Sum of subsets. Spanning trees – Prim's and Kruskal's algorithms, Single source shortest paths - Dijkstra's and Bellman Ford algorithms.