Essential Skills for 3rd Year Computer Engineering Students

In the 3rd year of a Computer Engineering program, students are expected to develop a combination of technical and
project-based skills to prepare for industry opportunities or advanced studies. The following guide provides a
comprehensive outline of the critical areas to focus on, each with its core theory, key topics, and suggested
practice questions to reinforce learning.

1. Programming and Data Structures

Theory: Study data structures such as arrays, linked lists, stacks, queues, trees, and graphs. Explore algorithmic concepts like sorting, searching, and recursion.

Key Topics:

- Types of data structures and applications
- Time and space complexity
- Key algorithms: quicksort, mergesort, binary search

Practice Questions: Implement each data structure, solve algorithm problems, practice sorting and searching.

2. Database Management and SQL

Theory: Learn relational databases, SQL, and NoSQL databases, covering database design, normalization, indexing, and queries.

Key Topics:

- SQL commands and queries (joins, subqueries, CTEs)
- Database normalization
- Basics of NoSQL (e.g., MongoDB)

Practice Questions: Design databases, write SQL queries, perform CRUD operations in both SQL and NoSQL contexts.

3. Operating Systems

Theory: Explore memory management, process scheduling, file systems, and concurrency in operating systems.

Key Topics:

- Process management, synchronization, and deadlock
- Memory management techniques
- File systems

Practice Questions: Describe scheduling algorithms, simulate simple memory allocation, explain file system structures.

4. Networking Basics

Theory: Cover the OSI model, TCP/IP protocols, IP addressing, and network security fundamentals.

Key Topics:

- OSI layers
- Protocols: TCP, UDP, HTTP, FTP
- Network troubleshooting

Practice Questions: Explain data flow in networks, troubleshoot network setups, analyze packets with tools like Wireshark.

5. Web Development

Theory: Learn HTML, CSS, JavaScript, React for front-end, and Node.js or Django for back-end development.

Key Topics:

- Front-end basics (HTML, CSS, JavaScript)
- Back-end development (APIs, servers)
- Connecting web applications to databases

Practice Questions: Build a small web application, create a responsive web page, design a CRUD application.

6. Cloud Computing Basics

Theory: Introduction to cloud services, virtualization, and using AWS, Azure, or GCP for scalable applications.

Key Topics:

- Cloud models (IaaS, PaaS, SaaS)
- Virtualization and containers (Docker)
- Core cloud services

Practice Questions: Set up a virtual server, deploy applications on the cloud, experiment with storage and compute services.

7. Machine Learning Basics

Theory: Introduction to machine learning, including regression, classification, clustering, and data pre-processing.

Key Topics:

- Supervised and unsupervised learning
- Model evaluation and training
- Libraries (TensorFlow, Scikit-Learn)

Practice Questions: Create a simple ML model, pre-process data, evaluate model accuracy.

8. Embedded Systems and IoT

Theory: Basics of microcontrollers (Arduino, Raspberry Pi), sensors, and IoT protocols.

Key Topics:

- Microcontrollers and sensors
- IoT connectivity and data handling
- Embedded programming

Practice Questions: Write code to interface with sensors, build a simple IoT project, simulate embedded applications.