First Term (4 Months)	
S01 - Structure Programming	S04 - Software Engineering
S01 - Object-Oriented Programming	S04 - Software Maintenance
S02 - Data Structure	S05 - Foundation of Basic Mathematics
S02 - Algorithm	
S03 - DBMS	
Second Term (4 Months)	
S06 – Fundamental Web Programming and Frameworks	S09 - Requirement
S06 – Front-End Development	S09 - Software Metrics
S07 – Back-End Development	S10 - Mathematics for AI, ML and DS
S07 - Full-Stack Development	
S08 - Database Design and Administration	
Third Term (4 Months)	
S11 - Virtualization and Cloud Computing	S14 - Project Management
S12 - Fundamental of DevOps Engineering	S14 - Testing and Quality Assurance
S12 - DevOps Tools	S15 - Statistics
S13 - Development Process	
S13 - UML and Documentation	
First Term (4 Months)	
S16 - Human-Computer Interaction	S20 - System Analysis and Design
S17 - Fundamental of Graphics Design	S20 - Software Security
S17 - Design Process	S21 - Numerical Analysis
S18 - UI Design	
S19 - UX Design	
Second Term (4 Months)	
S22 - Artificial Intelligence	S25 - Software Design and Analysis
S22 - Machine Learning	S25 - Architecture and Design Pattern
S23 - Pattern Recognizing and Image Processing	S25 - Software Architecture
S24 - Data Science, Big Data and Data Visualization	S26 - Combinatorial Optimization
S24 - Data Warehouse and Mining	
Third Term (4 Months)	
S27 - Theory of Computation	S32 - Business Psychology
S28 - Operating System and System Programming	S33 - Business Communication
S29 - Computer Network and Data Communication	S34 - Business Studies for Engineers
S30 - Computer Organization	S35 - English for Business Communication
S31 - Distributed System and Parallel Computing	S34 - Professional Ethics for Information System

Concept Level	Basic Key Concept
Basic Level	Basic Key Concept with Implementation
Intermediate Level	Intermediate Key Concept with Implementation
Professional Level	Professional Key Concept with Implementation
Expert Level	Expert Key Concept with Manipulation of Implementation
First 5 Days	GPT Notes of all the topics
	Real-Life Problem Solving Example, Questions and Quiz – 10 / 20
	Competitive Problem Solving Questions
	Interview Questions Similar as Google, Microsoft, OpenAI
	Make a Schedule of 5 Days
	Each day Create Summarized Note for each of the topics with the Implementation Practice
	All the Summarized Notes must be Categorized in Different Group
	Lines and Topics must be highlighted and Listed and Pointed
Last 2 Days	Review and Real-Life Implementation of all the Topics
	Solve all the Question and Quiz
	Market Analysis of the Product and Skills
	Knowledge Sharing with Community and Make Own Group

SL/NO	Part One	Engineering and Development (DSA & DB)
01	Structure Programming	Core Programming (C++ and JavaScript)
02	Data Structure	Core Programming (C++ and JavaScript)
03	Algorithm	Core Programming (C++ and JavaScript)
04	Database Management System	Database Design (MySQL and MongoDB)
05	Object-Oriented Programming	Core Programming (C++ and JavaScript)
	Part Two	**Engineering and Development (Architect)**
06	Software Engineering	
07	Software Design and Analysis	Micro-service, Scalability, Design Patterns – Data Intensive App
08	System Analysis and Design	
09	Software Security	
10	Professional Ethics for Information System	
	Part Three	**Engineering and Development (Web & AI)**
11	Web Technology and Frameworks	Web Development Basic (HTML, CSS, JavaScript)
	Backend Development	Node.JS and Express.JS
	Frontend Development	React.JS, State Management and Responsive Design
	Full Stack Development	API, Authentication (JWT|QAuth) and Advanced JavaScript
	Advanced Full Stack	Real-Time Apps (Web-socket) and Server-less Architecture
12	Artificial Intelligence and Machine Learning	
13	Applied Data Science and Engineering	
	Part Four	**Product Management**
14	Requirement Specification and Analysis	
15	Software Metrics	
16	Testing and Quality Assurance	Manual Testing and Testing Automation Tool (Selenium)
	QA Automation	Test Framework (Cypress|Appium), Perofrmance Testing
17	Project Management	Agile Methodology, Scrum, Stakeholder Management
	Part Five	**Product Management (DevOps Engineering)**
18	Software Maintenance	
19	Virtualization and Cloud Computing	
20	DevOps Fundamental	Linux Command Line, Version Control, CI/CD Fundamentals
	DevOps Advance	Docker, Kubernetes, Infrastructure as Code
21	Development Process	
	Part Six	**Design and User Experience**
22	Human-Computer Interaction	
23	UI/UX Design Fundamental	Design Principle, Figma and Prototyping
	UI/UX Design Advanced	Advance Prototyping, Usability Testing, Motion Design
24	Technical Writing and UML	

SL/NO	Part Six	Computer science (Mathematics)
24	Discrete Mathematics	
25	Numerical Analysis	
26	Probability and Statistics	
27	Calculus, Deferential Equation and Analytical Geometry	
28	Combinational Optimization	
	Part Seven	**Computer Science Part-1**
29	Theory of Computation	
30	Operating System and System Programming	
31	Computer Network	
32	Distributed System and Parallel Computing	
	Optional Group One	**Computer Science Part-2**
33	Computer Organization	
34	Computer Graphics and Multimedia	
35	Mobile and Wireless Computing	
36	Embedded System	
37	Pattern Recognizing and Image Processing	
	Optional Group Two	**Business Computing and Customer Support**
38	Numerical Computation for Financial Modeling	
39	Information Retrieval	
40	Enterprise Information System	
41	Data Mining and Warehouse	
42	Business Psychology	
43	Business Studies for Engineers	
44	Business Communication	CRM, Communication Strategies, Handling User Feedback
45	Strategic Management	

Here's a Bachelor of Science (BSc) degree-style curriculum table structured to cover Full Stack Development, Design, QA, DevOps, Project Management, Software Architecture, and Customer Support as Specialized areas. This schedule spans 8 semesters (4 years) and includes core courses, electives, projects, and industry-ready skills for expertise.

Semester	Subject Area	Topics	Learning Resources
01	Core Programming	Programming Fundamentals (Python, JavaScript), Algorithms, and Data Structures	- Introduction to the Theory of Computation by Michael Sipser - CS50's Introduction to Computer Science (Harvard)
02	Web Development Basics	HTML, CSS, JavaScript Basics	- HTML and CSS: Design and Build Websites by Jon Duckett - FreeCodeCamp Web Dev Guide
03	Backend Development	Node.JS, Express.JS, Database (SQL, MongoDB)	- Eloquent JavaScript by Marijn Haverbeke - MDN Backend Docs
04	UI/UX Design Basics	Design Principles, Figma, Prototyping	- The Elements of User Experience by Jesse James Garrett - Interaction Design Foundation
05	Frontend Development	React/Angular, State Management, Responsive Design	- Learning React by Kirupa Chinnathambi - Frontend Mastery by Codecademy
06	QA Testing Basics	Manual Testing, Introduction to Automation Testing Tools (Selenium)	- Testing Computer Software by Cem Kaner - Test Automation University
07	Full Stack Development	APIs, Authentication (JWT, OAuth), Advanced JavaScript	- The Odin Project Full Stack Path
08	UI/UX Advanced	Advanced Prototyping, Usability Testing, Motion Design	- Don't Make Me Think by Steve Krug - Design + Code Tutorials
09	Software Architecture	Micro-services, Design Patterns, Scalability	- Designing Data-Intensive Applications by Martin Kleppmann
10	Advanced Full Stack	Real-Time App (WebSocket), Server-less Architecture	- Node.JS in Action
11	QA Automation	Test Frameworks (Cypress, Appium), Performance Testing	- Continuous Testing for DevOps Professionals by Katrina Clokie
12	DevOps Basics	Linux Command Line, Git, CI/CD Fundamentals	- DevOps Full Course by Simplilearn
13	Project Management	Agile Methodology, Scrum, Stakeholder Management	- Scrum: The Art of Doing Twice the Work in Half the Time Agile M.
14	DevOps Advanced	Docker, Kubernetes, Infrastructure as Code	- The Phoenix Project by Gene Kim - Docker Documentation
15	Customer Support	CRM, Communication Strategies, Handling User Feedback	- Zendesk Customer Support Guide
16	Capstone Project	Build a Full-Scale Application Incorporating All Sills	- Mentorship Programs (linkedIn Learnig) - Personal GitHub Projects

Duration	Topics	Learning Resources	Practice Example	Collaboration Tools
Full Stack Development				
Week 1-4	HTML, CSS, JS Basic	- HTML & CSS by Jon Duckett - FreeCodeCamp	- Build a Portfolio Website - Frontend Mentor Challenges	- GitHub for version control - Discord for team discussions
Week 5-8	Backend (Node.JS, MongoDB)	- Eloquent JavaScript by Marijn Haverbeke - The Odin Project	- Build a REST API for a blog - API Practice	- GitLab for collaboration - Trello for task management
Week 9-12	Advanced Full Stack (React, Authentication)	- Learning React by Kirupa Chinnathambi - Scrimba React	- Build a real-time chat app - Socket.IO Demos	- VS Code Live Share for coding together
Design and User Experience				
Week 1-3	UI/UX Basics, Figma	- The Elements of User Experience by Jesse - Figma Tutorials	- Redesign a popular app's interface - Daily UI Challenges	- Figma Collaboration Tools - Miro for brainstorming
Week 4-5	Prototyping, User Research	- Don't Make Me Think by Steve Krug - User Research Basics	- Conduct a usability test for a basic prototype	- Optimal Workshop for usability testing
Week 6-8	Advanced Design (Motion, Accessibility)	- Google UX Design Certificate	- Create an accessible app interface - Contrast Checker	- XD Team Collaboration Features
Software Architecture				
Week 1-3	System Design Basic, Micro-services	- Designing Data-Intensive Applications by Martin Kleppmann	- Design an architecture for a social media platform	- Lucidchart or Draw.io for diagramming
Week 4-6	Scalability, Performance Optimization	- System Design Primer	- Optimize database queries	- AWS Architecture Tools
QA Engineering				
Week 1-2	Manual Testing Basics	- Testing Computer Software by Cem Kaner - ISTQB Foundations	- Test an e-commerce Website - Bug Reporting Practice	- Jira for test tracking - TestRail for test management
Week 3-6	Automation Testing (Selenium, Cypress)	- Test Automation University	- Write test cases for a web app - Selenium Project Ideas	- Browser-Stack for cross-browser testing
Week 7-8	Performance and Security Testing	- OWASP Testing Guide	- Load test with JMeter - Penetration test a small API	- OWASP ZAP for security testing
DevOps				
Week 1-2	CI/CD Basics, Git, Docker	- The Phoenix Project by Gene Kim - Docker Documentation	- Set up CI/CD with GitHub Actions - Create a Dockerized web app	- Jenkins for pipeline - Docker Hub for collaboration
Week 3-5	Kubernets, Infrastructure as Code	- Kubernetes Tutorials	- Deploy an app using Kubernetes - Practice with Terraform	- Kubernetes Dashboard
Week 6-8	Advanced Monitoring and Security	- Prometheus and Grafana Docs	- Monitor a live app - Visualize server performance	- Prometheuse and Grafana Tools
Project Management				
Week 1-2	Agile, Scrum Basics	- Scrum: The Art of Doing Twice the Work in Half the Time by Jeff Sutherland	- Plan a mock sprint with your team	- Trello/Asana for Agile project management