Assam University, Silchar

Four Year Undergraduate Programme

Implemented under NEP 2020

Effective from the Academic Year 2023-24

Syllabus of Computer Science
(3rd Year)

Approved in the 99th meeting of the Academic Council on 22.05.2025 vide Resolution No AC:99:05-25:5

Semester wise list of Computer Science Discipline Specific Core (DSC) Papers

SEMESTER	COURSE CODE	TITLE OF COURSES	CREDITS
I	**CSCDSC101**	Digital Computer Fundamentals	3
	CSCDSC102	Discrete Mathematics	3
II	**CSCDSC151**	Data Structure	3
	CSCDSC152	Lab on Data Structure	3
III	**CSCDSC201**	Computer Organization and Architecture	4
	CSCDSC202	Operating Systems	4
IV	**CSCDSC251**	Object Oriented Programming with Java	4
	CSCDSC252	Database Management System	4
	CSCDSC253	Lab on Java & DBMS	4
V	**CSCDSC301**	Computer Graphics	4
	CSCDSC302	System Analysis and Design & Cyber Security	4
	CSCDSC303	Lab on Graphics Programming	4
VI	**CSCDSC351**	Computer Network and Internet Technology	4
	CSCDSC352	Theory of Computation	4
	CSCDSC353	Microprocessor and Systems Programming	4
	CSCDSC354	Lab on Internet Technology & Microprocessor and Systems Programming / Project Work	4
VII	**CSCDSC401**	Design and Analysis of Computer Algorithms	4
	CSCDSC402	Principles of Compiler Design	4
	CSCDSC403	Artificial Intelligence	4
	CSCDSC404	Lab on DACA & Compiler Design	4
VIII	**CSCDSC451**	Software Engineering	4
	CSCDSC452(A)	Image Processing	4
	CSCDSC452(B)	Data Analytics	4
	CSCDSC453	Natural Language Processing	4
	CSCDSC454(A)	Internet of Things	4
	CSCDSC-454(B)	Cloud Computing	4
	OR		
	CSCDSC451	Research Methodology	4
	CSCDSC455	Research Project/Dissertation	12

Semester wise list of Computer Science Discipline Specific Minor (DSM) Papers

SEMESTER	COURSE CODE	TITLE OF COURSES	CREDITS	DSM1/DSM2
I	**CSCDSM101**	Programming with C	3	**DSM1**
II	**CSCDSM151**	Programming with C	3	**DSM2**
III	**CSCDSM201**	Database Management System	4	**DSM1**
IV	**CSCDSM251**	Lab on C & DBMS	3	**DSM1**
	CSCDSM252	Database Management System	3	**DSM2**
V	**CSCDSM301**	Operating Systems	3	**DSM1**
	CSCDSM302	Operating Systems	3	**DSM2**
VI	**CSCDSM351**	Lab on C & DBMS	4	**DSM2**
VII	**CSCDSM401**	Internet Technologies	4	**DSM1**
VIII	**CSCDSM451**	Internet Technologies	4	**DSM2**

Semester wise list of Computer Science Skill Enhancement Course (SEC) Papers

SEMESTER	COURSE CODE	TITLE OF COURSES	CREDITS
I	**CSCSEC101**	Programming with C	3
II	**CSCSEC151**	Python Programming	3
III	**CSCSEC201**	Programming with C++ & Lab on OS and C++	3

Semester wise list of Computer Science Interdisciplinary Course (IDC) Papers

SEMESTER	COURSE CODE	TITLE OF COURSES	CREDITS
I	**CSCIDC101**	Computer Fundamentals & Applications	3
II	**CSCIDC151**	Programming Fundamentals with C	3
III	**CSCIDC201**	Introduction to Web Designing & Cyber Security	3

Syllabi of Computer Science DSC Courses

Semester	**: V**
Course Type	**: DSC**
Course Code	**: CSCDSC301**
Name of the Course	**: Computer Graphics**
Learning level	**: Higher-level Course**
Credits	**: 4**
Contact Hours	**: 60**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

Course Objectives: *The course objective is to*

1. *Understand the basic principles of graphics algorithm design.*
2. *Familiarize the functionality of display devices.*
3. *Introduce how graphics hardware and software work in real work applications.*
4. *Understand the various algorithms for drawing structures and shapes.*
5. *Relate projection of an object in real world scenario.*

Unit-I

Introduction: Basic elements of Computer graphics, Applications of Computer Graphics.
Graphics Hardware: Architecture of Raster and Random scan display devices, input/output devices.

Unit-II

Fundamental Techniques in Graphics: Raster scan line, circle and ellipse drawing, thick primitives, Polygon filling, line and polygon clipping algorithms, inside-outside test, boundary and flood-fill.

Unit-III

2D transformation and viewing: Transformations (translation, rotation, scaling), matrix representation, homogeneous coordinates, composite transformations, reflection and shearing, viewing pipeline and coordinates system, window-to-viewport transformation, clipping including point clipping, line clipping (cohen-sutherland), polygon clipping.

Unit-IV

3D transformation and viewing: Bezier curves and surfaces, B-spline curves and surfaces. 3D scaling, rotation and translation, composite transformation, viewing pipeline and coordinates, parallel and perspective transformation, view volume and general (parallel and perspective) projection transformations.

Unit-V

Visible surface detection concepts: back-face detection, depth buffer method, illumination, light sources, illumination methods (ambient, diffuse reflection, specular reflection), Color models: properties of light, XYZ, RGB, YIQ and CMY color models. Concept of Animation and Morphing.

***Course Learning Outcomes**: After successful completion of the course, the students will be able to:*

1. *Apply output primitive's algorithm to create graphics.*
2. *Implement various algorithms to scan, convert the basic geometrical primitives, transformations, area filling, clipping.*
3. *Illustrate basic graphics application programs.*
4. *Perform basic projects in graphics.*
5. *Conceive the idea of 3D graphics & views.*
6. *Extend the application of graphics for other image processing*

Text Books:

1. D. Hearn, Baker: "**Computer Graphics**", Prentice Hall of India 2008.
2. D. F. Rogers: "**Procedural Elements for Computer Graphics**", McGraw Hill 1997.
3. J. D. Foley, A.Van Dan, Feiner, Hughes: "**Computer Graphics Principles & Practice**", 2nd Edition, Addison Wesley 1990.

Reference Books:

1. D. F. Rogers, Adams: "**Mathematical Elements for Computer Graphics"**, McGraw Hill 2nd edition 1989.
2. D. P. Mukherjee: "**Fundamentals Of Computer Graphics And Multimedia**", PHI Learning, 1st Edition.

Semester	**: V**
Course Type	**: DSC**
Course Code	**: CSCDSC302**
Name of the Course	**: System Analysis and Design & Cyber Security**
Learning level	**: Higher-level Course**
Credits	**: 4**
Contact Hours	**: 60**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

***Course Objectives:** The course objective is to*

1. *Equip students with the knowledge and skills necessary to analyze, design, develop, and implement information systems.*
2. *Understand System Development Life Cycle (SDLC), Requirement Analysis, Problem Solving & Decision Making, Software Development Approaches, User Interface Design, etc.*
3. *Understand project management principles and documentation in system development.*
4. *Learn the foundations of Cyber security and threat landscape.*
5. *Equip students with the technical knowledge and skills needed to protect and defend against cyber threats.*
6. *Develop skills in students that can help them plan, implement, and monitor cyber security mechanisms to ensure the protection of information technology assets.*
7. *Expose students to governance, regulatory, legal, economic, environmental, social and ethical contexts of cyber security.*
8. *Expose students to responsible use of online social media networks.*

9. *Systematically educate the necessity to understand the impact of cyber crimes and threats with solutions in a global and societal context.*

UNIT-I

Introduction to system Concept and analysis: Elements of a system, types of system. Introduction to system development life cycle, Recognition of need, prototyping, Roles of system Analyst, The analyst/ user interface. A**nalysis:** Determining the users information requirements, problem definition, Background analysis, fact-finding, fact analysis. Introduction to structured analysis, the tools of structured analysis, Feasibility study; oral representation, Data analysis, Cost/ Benefit analysis, the system proposal.

UNIT-II

Introduction to system designs and testing: The process of design (logical and physical design), Design methodology, structured design, structured walkthrough, Major development activities, Data validation. Introduction to input design, output design, forms design; File structure, File organization, Data Base design, and the role of DBA. **Testing:** The Test Plan, Quality assurance; System Conversion, Post implementation review, Software maintenance.
Procedure for Hardware/Software selection: Project Management and Control, project control, gantt chart, PERT and CPM.

UNIT-III

Introduction to Cyber security: Defining Cyberspace, Introduction to internet, web technology and www, Architecture of cyberspace, Regulation of cyberspace, Concept of cyber security, Issues and challenges of cyber security. **Cyber crime and Cyber law:** Classification of cyber crimes, Common cyber crimes- cyber crime targeting computers and mobiles, cyber crime against women and children, financial frauds, social engineering attacks, malware and ransomware attacks, zero day and zero click attacks, Cybercriminals modus-operandi , Reporting of cyber crimes, Remedial and mitigation measures, Legal perspective of cyber crime, IT Act 2000 and its amendments, Cyber crime and offences, Organisations dealing with Cyber crime and Cyber security in India, Case studies.

UNIT-IV

E-Commerce and Digital Payments: Definition of E- Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, E-Commerce security best practices, Introduction to digital payments, Components of digital payment and stake holders, Modes of digital payments- Banking Cards, Unified Payment Interface (UPI), e-Wallets, Unstructured Supplementary Service Data (USSD), Aadhar enabled payments, Digital payments related common frauds and preventive measures. RBI guidelines on digital payments and customer protection in unauthorised banking transactions. Relevant provisions of Payament Settlement Act,2007,

UNIT-V

Social Media Overview and Security: Introduction to Social networks. Types of Social media, Social media platforms, Social media monitoring, Hashtag, Viral content, Social media marketing, Social media privacy, Challenges, opportunities and pitfalls in online social network, Security issues related to social media, Flagging and reporting of inappropriate content, Laws regarding posting of inappropriate content, Best practices for the use of Social media, Case studies. **Digital Devices Security , Tools and Technologies for Cyber Security:** End Point device and Mobile phone security, Password policy, Security patch management, Data backup, Downloading and management

of third party software, Device security policy, Cyber Security best practices, Significance of host firewall and Ant-virus, Management of host firewall and Anti-virus, Wi-Fi security, Configuration of basic security policy and permissions.

***Course Learning Outcomes**: After successful completion of the course, the students will be able to:*

1. *Explain the importance and role of system analysis and design in information systems development.*
2. *Identify and gather system requirements from stakeholders using various techniques.*
3. *Create Data Flow Diagrams (DFD), Entity-Relationship Diagrams (ERD), and Use Case Diagrams to represent system models*
4. *Understand the concept of Cyber security and issues and challenges associated with it.*
5. *Understand the cyber crimes, their nature, legal remedies and as to how report the crimes through available platforms and procedures.*
6. *Appreciate various privacy and security concerns on online Social media and understand the reporting procedure of inappropriate content, underlying legal aspects and best practices for the use of Social media platforms.*
7. *Understand the basic concepts related to E-Commerce and digital payments. They will become familiar with various digital payment modes and related cyber security aspects, RBI guidelines and preventive measures against digital payment frauds.*
8. *Understand the basic security aspects related to Computer and Mobiles. They will be able to use basic tools and technologies to protect their devices.*

Text Book

1. Elias. M. Awad, **System Analysis and Design**, 2nd edition, Galgotia Publication, 1993.
2. Kenneth e. Kendall , julie e. Kendall, **systems analysis and design,** 7th edition, pearson publication
3. R. C Mishra**, Cyber Crime Impact in the New Millennium**, Auther Press. Edition 2010.
4. Sumit Belapure and Nina Godbole, **Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives**, Wiley India Pvt. Ltd. (1st Edition, 2011)
5. Henry A. Oliver**, Security in the Digital Age: Social Media Security Threats and Vulnerabilities**, Create Space Independent Publishing Platform. (Pearson, 13th November, 2001)

Reference Books:

1. Kendall and Kendall, **System analysis and Design**, PHI, 2006.
2. Igor Hawryszkiewycz, **Introduction to System analysis and Design**, PHI, 2010.

3. **Electronic Commerce** by Elias M. Awad, Prentice Hall of India Pvt Ltd.
4. **Cyber Laws: Intellectual Property & E-Commerce Security** by Kumar K, Dominant Publishers.
5. **Network Security Bible**, Eric Cole, Ronald Krutz, James W. Conley, 2nd Edition, Wiley India Pvt. Ltd.
6. **Fundamentals of Network Security** by E. Maiwald, McGraw Hill.

Semester : V
Course Type : DSC
Course Code : CSCDSC303
Name of the Course : Lab on Graphics Programming
Learning level : Higher-level Course
Credits : 4
Contact Hours : 60
Total Marks : 100
End Semester Marks : 70
Internal Marks : 30

Course Objectives: *The course objective is to*
1. Explain the basics of computer graphics and its applications.
2. Explain the methods to plot and trace various shapes like line, circle, ellipse etc.
3. Understand the implementation of various clipping and filling algorithms
4. Gain knowledge about practical implementations of 2D and 3D transformations and viewing.

This paper provides practical knowledge of Graphics Programming. List of laboratory programming assignments (not limited to these):

Lab on Graphics Programming

1. Write a C/C++ program to implement DDA line drawing algorithm
2. Write a C/C++ program to implement Bresenham's line drawing algorithm
3. Write a C/C++ program to implement mid-point circle drawing algorithm
4. Write a C/C++ program to implement the mid-point ellipse drawing algorithm.
5. Write a C/C++ program to implement Cohen Sutherland Line Clipping algorithm.
6. Write a C/C++ program to clip a polygon using Sutherland Hodgeman algorithm.
7. Write a C/C++ to implement Scan-Line Polygon Filling Algorithm.
8. Write a C/C++ program to apply various 2D transformations on a 2D object (use homogenous coordinates).
9. Write a C/C++ program to apply various 3D transformations on a 3D object and then apply parallel projection on it.
10. Write a C/C++ program to apply various 3D transformations on a 3D object and then apply perspective projection on it.
11. Write a C/C++ program to draw a circle of red colour inside of a rectangle of blue colour on a background of green colour.
12. Write a C/C++ program to draw Bezier curve.
13. Write a C/C++ program to draw B-spline curve.

***Course Learning Outcomes**: After successful completion of the course, the students will be able to:*

1. Write programs to display various shapes like line, rectangles, circles, ellipse etc.
2. Perform the transformations of objects using translation, scaling and rotation cases.
3. Perform transformations.
4. Create and execute graphical models.

Semester	**: VI**
Course Type	**: DSC**
Course Code	**: CSCDSC351**
Name of the Course	**: Computer Network and Internet Technology**
Learning level* **	**: Higher-level Course**
Credits	**: 4**
Contact Hours	**: 60**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

***Course Objectives:** The course objective is to*

1. *Understand the basic concepts, principles, and terminologies of computer networks and internet technology.*
2. *Develop an understanding of IP addressing (IPv4 & IPv6), subnetting, and network addressing schemes.*
3. *Explore the concepts of switching techniques used in modern networks.*
4. *Explore the architecture of the internet, DNS, HTTP/HTTPS, FTP, and cloud computing fundamentals.*
5. *Gain knowledge of web technologies including HTML, CSS, JavaScript, and web frameworks for website development.*

UNIT I

Introduction to Computer Networks: Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite; Analog and digital signal; Encoding schemes; Multiplexing techniques- FDM, TDM; Circuit switching and packet switching; transmission media.

UNIT II

Data Link Layer Functions and Protocol: Error detection and error correction techniques; data-link control- framing and flow control; Error recovery protocols- stop and wait ARQ, go-back-n ARQ; Aloha- pure and slotted, CSMA, CSMA/CD protocols; **Networks Layer Functions and Protocols: Logical addressing, subnetting and supernetting,** Routing; link state, distance vector, virtual private network.

UNIT III

Transport Layer Functions and Protocols : Transport services-error and flow control, Connection establishment and release- three way handshaking;

Overview of Application layer protocol: Overview of DNS protocol; domain name and their organisation, understanding the Internet protocol address, Client-server concept-

architecture and application, overview of WWW & HTTP protocol.

UNIT IV

Internet: structure of internet, history of internet; Internet protocol; Addressing in Internet; understanding the Internet protocol address, Client-server concept- architecture and application. **HTML:** HTML Basic, HTML tags, creating list in HTML, hyperlinks, multimedia, HTML forms, tables in HTML, style sheets in HTML; **XML:** XML-Introduction, Need for XML, Advantages.

UNIT V

Creating interactive and dynamic web pages with JavaScript: Client-side scripting languages, JavaScript overview; constants, variables, operators, expressions and statements; user-defined & built-in-functions; properties and methods of built-in objects, client-side form validation.

***Course Learning Outcomes**: After successful completion of the course, the students will be able to:*

1. *Explain the key concepts, components, and functions of computer networks and internet technologies.*
2. *Demonstrate knowledge of the OSI and TCP/IP models and their associated protocols such as IP, TCP, UDP, HTTP, DNS, and FTP.*
3. *Understand the web development tools along with different markup and scripting languages.*

Text Books:

1. B. A. Forouzan, **Data Communications and Networking**, 4th edition, THM, 2007.
2. S. Tanenbaum, **Computer Networks**, 4th edition, PHI , 2002
3. Laura Lemay, **Mastering HTML, CSS & Java Script web publishing**, BPB, 2016.
4. Thomas Powell, **HTML & CSS: The Complete Reference**, 5th Edition, TMH, 2017.

Reference Books:

1. Kurose, **Computer Networking**, 8th edition, Pearson, 2022.
2. Larry L. Peterson and Bruce S. Davie, **Computer Networks: A Systems Approach**, 6th Edition, Morgan Kaufmann Pub, 2021.
3. J. Jaworski, **Mastering JavaScript and Jscript**, BPB Publication, 2007.

Semester	**: VI**
Course Type	**: DSC**
Course Code	**: CSCDSC352**
Name of the Course	**: Theory of Computation**
Learning level	**: Higher-level Course**
Credits	**: 4**
Contact Hours	**: 60**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

Course Objectives: *The course objective is to*

1. *Introduce the theoretical foundations of computer science from the perspective of formal languages.*
2. *Formal methods of computation like automata theory, regular expressions and grammars*
3. *Define different types of formal language and their relationships.*
4. *Introduce the concept of Push Down Automata and the Turing Machine*

Unit-I

Languages and Finite Automata: Alphabets, string, language, Finite Automata, Transitions and its properties, Acceptability by Finite Automaton, Introduction to Nondeterministic Finite Automata, equivalence of NFA (Nondeterministic Finite Automata) and DFA (Deterministic Finite Automata), Minimization of finite Automata, Finite Automata with output -Mealy and Moore Machines.

Unit-II

Regular Sets and Regular Grammar: Formal definition, Kleene closure, algebra of regular expression, Regular languages, closure properties of regular set, finite automata and regular expressions

Unit-III

Context Free Languages: Context free grammars, parse trees, ambiguities in grammars and languages.

Unit-IV

Pushdown automata: Pushdown automata (Deterministic and Non-deterministic), Pumping Lemma, properties of context free languages, normal forms.

Unit-V

Chomsky Hierarchy and Turing Machines: Formal definition of Turing Machine, Transition diagram, Basic structure and working of Turing Machine, language of Turing Machine, Types of Turing Machine, universal Turing Machine, Chomsky Hierarchy.

Course Learning Outcomes*: After successful completion of the course, the students will be able to:*

1. *Understand the basic properties of formal languages and grammars.*
2. *Differentiate regular, context-free and context-sensitive languages.*
3. *Understand and implement grammars to produce strings from a specific language.*
4. *Understand concepts relating to Push Down Automata and Turing machines*

Text Books:

1. S. P. Eugene Xavier: "**Theory of Automata, Formal Languages and Computation**", New age International Publishers, 2005
2. Daniel I. A. Cohen: "**Introduction to Computer Theory"**, John Wiley, 1996
3. Hoperoft, Aho, Ullman: "**Introduction to Automata theory, Language & Computation**", 3rd Edition, Pearson Education, 2006

Reference Books:

1. Lewis & Papadimitriou: "**Elements of the Theory of Computation**", PHI, 1997.
2. P. Linz: "**An Introduction to Formal Language and Automata**", 5th edition, Jones and Bartlett Publishers, Inc., USA, 2012.

3. K. L. P. Misha, "Theory of Computer Science: Automata, Languages and Computation", PHI, 3rd Edition, 2006

.

Semester	**: VI**
Course Type	**: DSC**
Course Code	**: CSCDSC353**
Name of the Course	**: Microprocessor and Systems Programming**
Learning level	**: Higher-level Course**
Credits	**: 4**
Contact Hours	**: 60**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

Course Objectives: *The course objective is to*

1. Understand the various functional units of computer.
2. Understand the architecture of 8086 Microprocessor
3. Understand about various types of addressing modes.
4. Learn the various instructions set of 8086 processor, so that they will be able to do assembly language programming.
7. Understand Basic architecture of computer system programs.
8. Learn how high level language converted to machine level language

Unit-I

Microprocessor Architecture: Microprocessor Architecture and its Operations, Fetching, decoding and execution of an Instruction. Instruction Set of 8086: 8086 Programming Model, Instruction Classification, Instruction and Data Format, Addressing Modes, Memory interfacing to 8086. Data Transfer Operations, Arithmetic Operations, Logic Operations, Branch Operations.

Unit-II

Programming Techniques: Looping, Counting and Indexing, Additional Data Transfer and 16-bit Arithmetic Instructions, Arithmetic Operation related to Memory, Logic Operations: Rotate, Compare, Counters and Time Delays with few examples. Stacks and Subroutines: Stack, Subroutine, Restart, conditional call, and return instructions; BCD Addition, BCD Subtraction, Introduction to advanced instructions and applications, multiplication, subtraction with carry.

Unit-III

Interrupts & Interfacing Data Converters: 8086 Interrupt, 8086 Vectored Interrupts, Direct Memory Access, Digital-to-Analog Converter - basic concepts, Interfacing 8-bit D/A Converter, Analog-to-Digital Converter - basic concepts, Interfacing 8-bit A/D Converter.

Unit-IV

Overview of System Software and Assemblers: Introduction, Software Hierarchy, Systems software and Machine Architecture, Symbol Tables, Elements of Assembly Language Programming, Design of the Assembler, Assembler Design Criteria, Types of Assemblers, Two-Pass Assemblers, One-Pass Assemblers, Algorithm of Single Pass Assembler, Multi-Pass Assemblers, Macro Definition and Expansion.

Unit-V

Linkers and Loaders: Introduction, Relocation of Linking Concept, Design of a Linker, Self Relocating Programs, Dynamic Linking, Loaders, Different Loading Schemes, Sequential and Direct Loaders, Compile-and-Go Loaders, General Loader Schemes, Absolute Loaders, Relocating Loaders, Linking Loaders, Relocating Linking Loaders.

Course Learning Outcomes*: After successful completion of the course, the students will be able to:*

1. Familiar with functional units of CPU.
2. Aware about architecture of 8086 Microprocessor.
3. Understand about various types of instruction formats and addressing modes.
5. Perform different operations on data using assembly language programming.
6. Familiar with the concept of Interfacing of I/O and Memory peripherals with microprocessor.
7. Understand the basic concepts of system software and the importance of system program.
8. Learn to write assembly level program code and the conversion into the machine level code.
9. Learn the dynamic allocation of memory to program code and linking and loading these program codes.

Text Books:

1. Ramesh S. Gaonkar, "**Microprocessors Architecture, Programming and Applications**", Wiley eastern, 6th edition, 2013.
2. A. K. Ray and K. M. Bhurchandi, "**Advanced microprocessor and Peripherals"**, TMH, 2000.
2. John J. Donovan, "**Systems Programming**", Tata Mc Graw Hill, 1st edition, 2017.
3. Leland. L. Beck, "**System Software**", Pearson Education, 3rd edition, 2007.

Reference Books:

1. Douglas U. Hall, "**Micro Processors & Interfacing**", Tata Mc Graw Hill, 3rd edition, 2017.
2. Aditya P. Mathur, "**Introduction to microprocessors**", TMH, 3rd edition 2018.

Semester	**: VI**
Course Type	**: DSC**
Course Code	**: CSCDSC354**
Name of the Course	**: Lab on Internet Technology & Microprocessor and Systems Programming**
Learning level	**: Higher-level Course**
Credits	**: 4**
Contact Hours	**: 60**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

Course objectives: *The course objective is to*

1. *Provide an in-depth understanding of the operations of microprocessors, assembly language programming and interfacing.*
2. *Provide an in-depth understanding of the operations of basic microcontrollers.*

This paper provides practical knowledge of Internet Technology & Microprocessor and Systems Programming. List of laboratory programming assignments (not limited to these):

Lab on Internet Technology

Create event driven program for following:
1. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
2. Print the largest of three numbers.
3. Find the factorial of a number n.
4. Enter a list of positive numbers terminated by Zero. Find the sum and average of these numbers.
5. A person deposits Rs 1000 in a fixed account yielding 5% interest. Compute the amount in the
account at the end of each year for n years.
6. Read n numbers. Count the number of negative numbers, positive numbers and zeros in the list.

Lab on Microprocessor and Systems Programming

1. Write an assembly language Program to perform different modes of operation (HEX KEYPAD/ Serial modes).
2. Write an assembly language Program to examine a series of memory byte locations from 0:1234.
3. Write an assembly language Program to examine and modify memory word location from 500:340.
4. Write an assembly language Program to examine registers.
5. Write an assembly language Program to modify registers DX to 55AA
6. Executing programs stored at memory location 1000H
7. Write an assembly language Program to convert a hexadecimal number to decimal number.
8. Write an assembly language Program to convert decimal to hexadecimal numbers.
9. Write an assembly language Program for hexadecimal addition/subtraction
10. Write an assembly language Program for hexadecimal multiplication/division

Course Learning Outcomes: *After successful completion of the course, the students will be able to:*

1. Understand the architecture and working principles of microprocessors.
2. Write a program in assembly level language.
3. Acquire skills in memory and peripheral interfacings.
4. Gain knowledge of microcontrollers and applications.
5. Solve real-world problems.

* The Department may offer a "Project Work" in lieu of "Lab on Internet Technology & Microprocessor and Systems Programming" for the students under allocated Project Supervisors as per the existing rules of Assam University, Silchar. (AC Resolution: 99: 05-25:26.8)

Syllabi of Computer Science DSM Courses

Semester : **V**
Course Type : **DSM**
Course Code : **CSCDSM301**
Name of the Course : **Operating Systems**
Learning level : **Higher-level Course**
Credits : **3**
Contact Hours : **45**
Total Marks : **100**
End Semester Marks : **70**
Internal Marks : **30**

Course Objectives: *The course objective is to*

2. *Understand the fundamental concepts of operating systems, including processes, threads, scheduling, etc.*
3. *Learn about the structure and components of operating systems, such as the kernel, device drivers, system calls, and user interface layers.*
4. *Explore process management concepts, including process creation, scheduling algorithms, inter-process communication, synchronization mechanisms, and deadlock handling.*
5. *Understand memory management techniques, including virtual memory, paging, segmentation, memory allocation algorithms*
6. *Explore operating system security mechanisms, including authentication, authorization, access control, encryption, and security policies*

Unit I

Introduction: Basic OS functions, resource abstraction, types of operating systems– multiprogramming systems, batch systems, time sharing systems; operating systems for personal Computers & workstations, process control & real time systems.
Operating System Organization: Processor and user modes, kernels, system calls and system programs.

Unit II

Process Management: System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-pre-emptive and preemptive scheduling algorithms.

Unit III

Process Coordination: Synchronization, concurrent processes, critical section, semaphores, methods for inter-process communication; deadlocks.

Unit IV

Memory Management: Physical and virtual address space; memory allocation strategies – fixed and variable partitions, paging, segmentation, virtual memory.

Unit V

File and I/O Management: Directory structures, file operations, file allocation methods, device management.

Protection and Security: Policy mechanism, Authentication, Internal access Authorization.

Course Learning Outcomes*: After successful completion of the course, the students will be able to:*

1. *Understanding of Operating System Concepts including processes, threads, memory management, file systems, and I/O operations.*
2. *Skills to analyse and solve problems related to process scheduling, resource allocation, etc.*
3. *Develop the ability to implement basic operating system functionalities such as process management, memory management, and file system operations.*
4. *Proficiency in file system organization, file I/O operations, directory management, and file allocation methods.*

Text Books:

1. A. Silberschatz, P.B. Galvin, G. Gagne, **Operating Systems Concepts**, John Wiley Publications. 8th Edition, 2008.
2. A. S. Tanenbaum, **Modern Operating Systems**, Pearson Education, 3rd Edition, 2007.

Reference Books:

1. G. Nutt, **Operating Systems: A Modern Perspective**, Pearson Education, 2nd Edition 1997.
2. W. Stallings, **Operating Systems, Internals & Design Principles**, PHI, 5th Edition, 2008.

Semester	**: V**
Course Type	**: DSM**
Course Code	**: CSCDSM302**
Name of the Course	**: Operating Systems**
Learning level	**: Higher-level Course**
Credits	**: 3**
Contact Hours	**: 45**
Total Marks	**: 100**
End Semester Marks	**: 70**
Internal Marks	**: 30**

Course Objectives: *The course objective is to*

7. *Understand the fundamental concepts of operating systems, including processes, threads, scheduling, etc.*
8. *Learn about the structure and components of operating systems, such as the kernel, device drivers, system calls, and user interface layers.*

9. *Explore process management concepts, including process creation, scheduling algorithms, inter-process communication, synchronization mechanisms, and deadlock handling.*
10. *Understand memory management techniques, including virtual memory, paging, segmentation, memory allocation algorithms*
11. *Explore operating system security mechanisms, including authentication, authorization, access control, encryption, and security policies*

Unit I

Introduction: Basic OS functions, resource abstraction, types of operating systems– multiprogramming systems, batch systems, time sharing systems; operating systems for personal Computers & workstations, process control & real time systems.
Operating System Organization: Processor and user modes, kernels, system calls and system programs.

Unit II

Process Management: System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-pre-emptive and preemptive scheduling algorithms.

Unit III

Process Coordination: Synchronization, concurrent processes, critical section, semaphores, methods for inter-process communication; deadlocks.

Unit IV

Memory Management: Physical and virtual address space; memory allocation strategies – fixed and variable partitions, paging, segmentation, virtual memory.

Unit V

File and I/O Management: Directory structures, file operations, file allocation methods, device management.
Protection and Security: Policy mechanism, Authentication, Internal access Authorization.

Course Learning Outcomes*: After successful completion of the course, the students will be able to:*

1. *Understanding of Operating System Concepts including processes, threads, memory management, file systems, and I/O operations.*
2. *Skills to analyse and solve problems related to process scheduling, resource allocation, etc.*
3. *Develop the ability to implement basic operating system functionalities such as process management, memory management, and file system operations.*
4. *Proficiency in file system organization, file I/O operations, directory management, and file allocation methods.*

Text Books:

1. A. Silberschatz, P.B. Galvin, G. Gagne, **Operating Systems Concepts**, John Wiley Publications. 8th Edition, 2008.
2. A. S. Tanenbaum, **Modern Operating Systems**, Pearson Education, 3rd Edition, 2007.

Reference Books:

1. G. Nutt, **Operating Systems: A Modern Perspective**, Pearson Education, 2nd Edition 1997.
2. W. Stallings, **Operating Systems, Internals & Design Principles**, PHI, 5th Edition, 2008.

Semester : VI
Course Type : DSM
Course Code : CSCDSM351
Name of the Course : Lab on C & DBMS
Learning level : Higher-level Course
Credits : 4
Contact Hours : 60
Total Marks : 100
End Semester Marks : 70
Internal Marks : 30

Course objective: *To provide students with practical skills and experiences in C programming and DBMS application*

1. *Develop proficiency in C programming language by implementing various programming tasks.*
2. *Understand memory management concepts in C, including dynamic memory allocation.*
3. *Gain a basic understanding of Database Management System (DBMS) concepts.*
4. *Understand the basics of Structured Query Language (SQL) for database querying and manipulation, including SELECT, INSERT, UPDATE, DELETE statements, and simple SQL queries.*

This paper provides practical knowledge of C Programming and SQL queries. List of laboratory programming assignments (not limited to these):

Lab on C Programming

1. Write a program to

   a) To check whether a number is prime or not

   b) To check whether a number is even or not

   c) Produce ASCII equivalent of given number

   d) Find divisor or factorial of a given number

   e) Find largest and smallest element in an array

f) Find sum of a geometric series

g) Cipher a string

h) Check whether a given string follows English capitalization rules

i) Find sum of the following series 1+ ½ + 1/3 +_____+1/20

j) Search whether a given substring exist in an input string or not and then delete this string from input string

2. Write a recursive program for tower of Hanoi problem.
3. The Fibonacci sequence of numbers is 1, 1, 2, 3, 5, 8……. Based on the recurrence relation F(*n*)=F(*n*-1)+F(*n*-2) for n>2 Write a recursive program to print the first *n* Fibonacci number.
4. Write a menu driven program for matrices to do the following operation depending on whether the operation requires one or two matrices
   a) Addition of two matrices
   b) Subtraction of two matrices
   c) Finding upper and lower triangular matrices
   d) Trace of a matrix
   e) Transpose of a matrix
   f) Check of matrix symmetry
   g) Product of two matrices.
5. Write a program that takes two operands and one operator from the user perform the operation and then print the answer
7. Write functions to add, subtract, multiply and divide two complex numbers (x+iy) and (a+ib) Also write the main program.
8. Write a menu driven program for searching an sorting with following options:-
a) Searching (1) Linear searching (2) Binary searching
b) Sorting (1) Insertion sort (2) Selection sort
9. Write a program to copy one file to other, use command line arguments.
10. Write a program to mask some bit of a number (using bit operations)
11. An array of record contains information of managers and workers of a company.

Print all the data of managers and workers in separate files.

12. Write a program to Evaluate the following algebraic expressions after reading necessary values from the user (ax+b)/ (ax-b) -2.5 log x-cos 30+|x2-y2|+sqrt (2xy) -(x5+10x4+8x3+4x+2)

Lab on DBMS

1. Implementation of DDL commands of SQL with suitable examples a) Create table b) Alter table c) Drop Table.
2. Implementation of DML commands of SQL with suitable examples a) Select b) Insert c) Update d) Delete.
3. Implementation of different types of function with suitable examples a) Number function b) Aggregate Function c) Character Function d) Conversion Function e) Date Function.
4. Implementation of different types of operators in SQL a) Arithmetic Operators b) Logical Operators c) Comparison Operator d) Special Operator e) Set Operation.

5. Implementation of different types of Joins a) Inner Join b) Outer Join c) Natural Join etc.
6. Study and Implementation of a) Group By & having clause b) Order by clause c) Indexing.
7. Study & Implementation of a) Sub queries b) Views.
8. Study & Implementation of different types of constraints.
9. Study & Implementation of Database Backup & Recovery commands. Study & Implementation of Rollback, Commit, Savepoint.
10. Creating Database /Table Space a) Managing Users: Create User, Delete User b) Managing roles:-Grant, Revoke.

Course Learning Outcomes: *After successful completion of the course, the students will be able to.*

1. *Demonstrate proficiency in programming using the C language, including understanding of syntax, data types, control structures, functions, and pointers.*
2. *Utilize file handling techniques in C for reading from and writing to files, including text and binary files, and perform input/output operations effectively.*
3. *Understand memory management concepts in C, including dynamic memory allocation*
4. *Understand the basics of Structured Query Language (SQL) for database querying and manipulation, including SELECT, INSERT, UPDATE, DELETE statements, and simple SQL queries.*