Date :- 25/04/2025

NOTIFICATION

Sub :- CBCS Syllabi of B. Sc. in Comp. Scie (Sem. III & VI)

Ref. :- Decision of the Academic Council at its meeting held on 22/04/2025.

The Syllabi of B. Sc. in Comp. Scie (Third and Fourth Semesters) as per **NATIONAL EDUCATION POLICY – 2020 (2024 Pattern)** and approved by the Academic Council as referred above are hereby notified for implementation with effect from the academic year 2025-26.

Copy of the Syllabi Shall be downloaded from the College Website (www.kcesmjcollege.in)

Sd/-
Chairman,
Board of Studies

To :

1) The Head of the Dept., M. J. College, Jalgaon.
2) The office of the COE, M. J. College, Jalgaon.
3) The office of the Registrar, M. J. College, Jalgaon.

Near Samarth Colony, Jalgaon - 425 002. (Maharashtra, India) | Ph. : +91 257 - 2237363, 2234281 | e-mail : mjcollege@kces.in | Website : mjcollege.kces.in
समर्थ कॉलनी जवळ, जळगांव – ४२५ ००२. (महाराष्ट्र, भारत) । दूरध्वनी क्र. : +९१ २५७–२२३७३६३, २२३४२८१ । ई–मेलः mjcollege@kces.in । वेबसाईटः mjcollege.kces.in

Khandesh College Education Society's

Moolji Jaitha College, Jalgaon

An "Autonomous College"

Affiliated to

Kavayitri Bahinabai Chaudhari

North Maharashtra University, Jalgaon-425001

KNOWLEDGE IS POWER

ESTD. 1945

STRUCTURE AND SYLLABUS

B.Sc. Honours/Honours with Research
(S.Y. B.Sc. Computer Science)

Under Choice Based Credit System (CBCS)
and
as per NEP-2020 Guidelines

[w.e.f. Academic Year: 2025-26]

Preface

The main aim of this course is to develop the technical skills of the candidate and make them experts in the process of using computers and other computer-related technologies. The course initially educates the candidates about all the basic fundamentals of the computer to strengthen their core so that they will find it easy to understand other complicated subjects that are there in the course.

Candidates with a graduate degree in Computer Science are getting a lot of job opportunities from different sectors. Businesses are using new advanced technologies for the smooth running of different operations, but with more advanced technology, different functions are becoming more sophisticated and complicated, this is when organizations need a Computer Science graduate to make this upgrading procedure simple and easy ensuring the smooth running of the business.Companies are hiring more Computer Science graduates in recent years to deal with the complex and complicated evolving of different technologies. A Computer Science graduate makes sure that all the computer facilities of the business are properly managed and handled by the workforce.

This course creates IT professionals who have done expertise in dealing with different technical problems. This course develops and upgrades the candidate's skills in different IT areas with term exams, practical learning, arranging summer internships for giving an IT workforce experience, and also organizing seminars with IT professionals so that the candidates can interact with the experts and clarify their doubts.
Hence, Board of Studies in Computer Science in its meeting held on 22nd March 2025 resolved to accept therevised syllabus for S. Y. B. Sc. (Computer Science) based on Choice Based Credit System (CBCS) ofUGC, NEP-2020 and the Government of Maharashtra guidelines.

Program Outcomes (PO) for B.Sc. Program:

Program Outcomes associated with a B.Sc. degree are as follows:

PO No.	PO
1	Graduates should have a comprehensive knowledge and understanding of the fundamental principles, theories, and concepts in their chosen field of study.
2	Graduates should possess the necessary technical skills and competencies related to their discipline, including laboratory techniques and data analysis.
3	Graduates should be able to identify, analyze, and solve complex problems using logical and critical thinking skills. They should be able to apply scientific methods and principles to investigate and find solutions.
4	Graduates should be proficient in effectively communicating scientific information, both orally and in writing.
5	Graduates should have a basic foundation in research methods and be capable of designing and conducting scientific investigations.
6	Graduates should be able to work effectively as part of a team, demonstrating the ability to collaborate with others, respect diverse perspectives, and contribute to group projects.
7	Graduates should recognize the importance of ongoing learning and professional development. They should be equipped with the skills and motivation to engage in continuous learning, adapt to new technologies and advancements in their field, and stay updated with current research.

Programme Specific Outcome (PSO) for B.Sc. Computer Science **Honours/Honours with Research:**

After completion of this program, students are expected to learn/understand the:

PSO No.	PSO
1	Database Management System and store the data in adequate form.
2	Object oriented programming languages such as C++ , Java Programming.
3	Various web technologies used for website development.
4	Core concepts in computer science
5	Different front end framework tools
6	Latest technologies such as IoT and Data Analytics.

Multiple Entry and Multiple Exit options:

The multiple entry and exit options with the award of UG certificate/ UG diploma/ or three-year degree depending upon the number of credits secured;

Levels	Qualification Title	Credit Requirements		Semester	Year
		Minimum	Maximum		
4.5	UG Certificate	40	44	2	1
5.0	UG Diploma	80	88	4	2
5.5	Three Year Bachelor's Degree	120	132	6	3
6.0	Bachelor's Degree- Honours **Or** Bachelor's Degree- Honours with Research	160	176	8	4

Credit distribution structure for Three/ Four year Honors/ Honors with Research Degree Programme with Multiple Entry and Exit

F.Y. B.Sc.

Year (Level)	Sem	Subject-I (M-1)	Subject-II (M-2)	Subject-III (M-3)	Open Elective (OE)	VSC, SEC (VSEC)	AEC, VEC, IKS	CC, FP, CEP, OJT, RP	Cumulative Credits/Sem	Degree/ Cumulative Credit
I (4.5)	I	DSC-1(2T) DSC-2(2P)	DSC-1(2T) DSC-2(2P)	DSC-1(2T) DSC-2(2P)	OE-1(2T)	---	AEC-1(2T) (Eng) VEC-1(2T) (ES) IKS(2T)	CC-1(2T)	22	**UG Certificate**
	II	DSC-3(2T) DSC-4(2P)	DSC-3(2T) DSC-4(2P)	DSC-3(2T) DSC-4(2P)	OE-2(2T) OE-3(2P)	---	AEC-2(2T) (Eng) VEC-2(2T) (CI)	CC-2(2T)	22	
	Cum. Cr.	**8**	**8**	**8**	**6**	**---**	**10**	**4**	**44**	
Exit option: Award of UG Certificate with 44 credits and an additional 4 credits core NSQF course/ Internship OR Continue with Major and Minor.										

S.Y. B.Sc.

Year (Level)	Sem	Subject-I (M-1) Major*		Subject-II (M-2) Minor #	Subject-III (M-3)	Open Elective (OE)	VSC, SEC (VSEC)	AEC, VEC, IKS	CC, FP, CEP, OJT/Int/RP	Cumulative Credits/Sem	Degree/ Cumulative Credit
		Mandatory (DSC)	Elective (DSE)	(MIN)							
2 (5.0)	III	DSC-5(2T) DSC-6(2T) DSC-7(2P)	---	MIN-1(2T) MIN-2(2T) MIN-3(2P)	---	OE-4(2T)	SEC-1(2T)	AEC-3(2T) (MIL)	CC-3(2T) CEP(2)	22	UG Diploma
	IV	DSC-8(2T) DSC-9(2T) DSC-10(2P)	---	MIN-4(2T) MIN-5(2P)	---	OE-5(2T)	SEC-2(2T) SEC-3(2P)	AEC-4(2T) (MIL)	CC-4(2T) ۞FP(2)	22	
	Cum. Cr.	**12**	**---**	**10**	**---**	**4**	**6**	**4**	**8**	**44**	
	Exit option: Award of UG Diploma in Major and Minor with 88 credits and an additional 4 credits core NSQF course/ Internship OR Continue with Major & Minor.										

***** Student must choose one subject as a Major subject out of M-1, M-2 and M-3 that he/she has chosen at First year
#Student must choose one subject as a Minor subject out of M-1, M-2 and M-3 that he/she has chosen at First year (Minor must be other than Major)
۞ OJT/Internship/CEP should be completed in the summer vacation after 4th semester

T.Y. B.Sc.

Year (Level)	Sem	Subject-I (M-1) Major		Subject-II (M-2) Minor	Subject-III (M-3)	Open Elective (OE)	VSC, SEC (VSEC)	AEC, VEC, IKS	CC, FP, CEP, OJT/Int/RP	Cumulative Credits/Sem	Degree/ Cumulative Credit
		Mandatory (DSC)	Elective (DSE)	(MIN)							
3 (5.5)	V	DSC-11(2T) DSC-12(2T) DSC-13(2T) DSC-14(2P) DSC-15(2P)	DSE-1A/B (2T) DSE-2A/B (2P)	---	---	---	VSC-1(2T) VSC-2(2P)	---	OJT/Int (4)	22	UG Degree
	VI	DSC-16(2T) DSC-17(2T) DSC-18(2T) DSC-19(2T) DSC-20(2T) IKS DSC-21(2P) DSC-22(2P)	DSE-3A/B (2T) DSE-4A/B (2P)	---	---	---	VSC-3(2T) VSC-4(2P)	---	---	22	
	Cum. Cr.	**24**	**8**	**---**	**---**	**---**	**8**	**---**	**4**	**44**	
	Exit option: Award of UG Degree in Major with 132 credits OR Continue with Major and Minor										

Fourth Year B.Sc. (Honours)

Year (Level)	Sem	Major Core Subjects		Research Methodology (RM)	VSC, SEC (VSEC)	OE	AEC, VEC, IKS	CC, FP, CEP, OJT/Int/RP	Cumulative Credits/Sem	Degree/ Cumulative Credit
IV (6.0)	VII	DSC-23(4T) DSC-24(4T) DSC-25(4T) DSC-26(2P)	DSE-5A/B (2T) DSE-6A/B (2P)	RM(4T)	---	---	---	---	22	UG Honours Degree
	VIII	DSC-27(4T) DSC-28(4T) DSC-29(4T) DSC-30(2P)	DSE-7A/B (2T) DSE-8A/B (2P)	---	---	---	---	OJT/Int (4)	22	
	Cum. Cr.	**28**	**8**	**4**	**---**	**---**	**---**	**4**	**44**	
Four Year UG Honors Degree in Major and Minor with 176 credits										

Fourth Year B.Sc. (Honours with Research)

Year (Level)	Sem	Major Core Subjects		Research Methodology (RM)	VSC, SEC (VSEC)	OE	AEC, VEC, IKS	CC, FP, CEP, OJT/Int/RP	Cumulative Credits/Sem	Degree/ Cumulative Credit
IV (6.0)	VII	DSC-23(4T) DSC-24(4T) DSC-26(2P)	DSE-5A/B (2T) DSE-6A/B (2P)	RM(4T)	---	---	---	RP(4)	22	UG Honours with Research Degree
	VIII	DSC-27(4T) DSC-28(4T) DSC-30(2P)	DSE-7A/B (2T) DSE-8A/B (2P)	---	---	---	---	RP(8)	22	
	Cum. Cr.	**20**	**8**	**4**	**---**	**---**	**---**	**12**	**44**	
Four Year UG Honours with Research Degree in Major and Minor with 176 credits										

Sem- Semester, **DSC**- Department Specific Course, **DSE**- Department Specific Elective, **OE/GE**- Open/Generic elective, **VSC**- Vocational Skill Course, **SEC**- Skill Enhancement Course, **VSEC**- Vocation and Skill Enhancement Course, **AEC**- Ability Enhancement Course, **IKS**- Indian Knowledge System, **VEC**- Value Education Course, **T**- Theory, **P**- Practical,**CC**-Co-curricular **RM**- Research Methodology, **OJT**- On Job Training, **FP**- Field Project, **Int**- Internship, **RP**- Research Project, **CEP**- Community Extension Programme, **ENG**- English, **CI**- Constitution of India, **MIL**- Modern Indian Laguage

- Number in bracket indicate credit
- The courses which do not have practical 'P' will be treated as theory 'T'
- If student select subject other than faculty in the subjects M-1, M-2 and M-3, then that subject will be treated as Minor subject, and cannot be selected as Major at second year.

Details of S.Y. B.Sc. (Computer Science)

Course	Course Type	Course Code	Course Title	Credits	Teaching Hours/ Week			Marks			
					T	P	Total	Internal		External	
								T	P	T	P
Semester III, Level – 5.0											
DSC-5	DSC	CS-DSC-231	Data Structures-I	2	2	---	2	20	---	30	---
DSC-6	DSC	CS-DSC-232	Java Programming-I	2	2	---	2	20	---	30	---
DSC-7	DSC	CS-DSC-233	Practical on Data structures and Java Programming-I	2	---	4	4	---	20	---	30
SEC-1	SEC	CS-SEC-231	HTML 5.0-I	2	2	---	2	20	---	30	---
CEP	CEP	CS-CEP-231	Community Engagment Program	2	---	4	4	50	---	---	---
MIN-1	MIN	CS-MIN-231	Programming in C-I	2	2	---	2	20	---	30	---
MIN-2	MIN	CS-MIN-232	Word Processig Tools	2	2	---	2	20	---	30	---
MIN-3	MIN	CS-MIN-233	Practical on Programming in C-I	2	---	4	4	---	20	---	30
OE-4	OE	CS-OE-231	Professional Presentation Skills	2	2	---	2	20	---	30	---
Semester IV, Level – 5.0											
DSC-8	DSC	CS-DSC-241	Data Structures-II	2	2	---	2	20	---	30	---
DSC-9	DSC	CS-DSC-242	Java Programming-II	2	2	---	2	20	---	30	---
DSC-10	DSC	CS-DSC-243	Practical on Data structures and Java Programming-II	2	---	4	4	---	20	---	30
SEC-2	SEC	CS-SEC-241	HTML 5.0-II	2	2	---	2	20	---	30	---
SEC-3	SEC	CS-SEC-242	Practical on HTML 5.0	2	---	4	4	---	20	---	30
FP	FP	CS-FP-241	Field Project	2	---	4	4	50	---	---	---
MIN-4	MIN	CS-MIN-241	Programming in C-II	2	2	---	2	20	---	30	---
MIN-5	MIN	CS-MIN-242	Practical on Programming in C-II	2	---	4	4	---	20	---	30
OE-5	OE	CS-OE-241	Basics of Internet	2	2	---	2	20	---	30	---

Examination Pattern

Theory Question Paper Pattern:

- 30 (External) +20 (Internal) for 2 credits
    - External examination will be of 1½ hours duration
    - There shall be 3 questions: Q1 carrying 6 marks and Q2, Q3 carrying 12 marks each. The tentative pattern of question papers shall be as follows;
    - Q1 Attempt any 2 out of 3 sub-questions; each 3 marks
    - Q 2 and Q3 Attempt any 3 out of 4 sub-question; each 4 marks.

Rules of Continuous Internal Evaluation:

The Continuous Internal Evaluation for theory papers shall consist of two methods:

1. Continuous & Comprehensive Evaluation (CCE): CCE will carry a maximum of 30% weightage (30/15 marks) of the total marks for a course. Before the start of the academic session in each semester, the subject teacher should choose any three assessment methods from the following list, with each method carrying 10/5 marks:

i. Individual Assignments
ii. Seminars/Classroom Presentations/Quizzes
iii. Group Discussions/Class Discussion/Group Assignments
iv. Case studies/Case lets
v. Participatory & Industry-Integrated Learning/Field visits
vi. Practical activities/Problem Solving Exercises
vii. Participation in Seminars/Academic Events/Symposia, etc.
viii. Mini Projects/Capstone Projects
ix. Book review/Article review/Article preparation
x. Any other academic activity
xi. Each chosen CCE method shall be based on a particular unit of the syllabus, ensuring that three units of the syllabus are mapped to the CCEs.

2. Internal Assessment Tests (IAT): IAT will carry a maximum of 10% weightage (10/5 marks) of the total marks for a course. IAT shall be conducted at the end of the semester and will assess the remaining unit of the syllabus that was not covered by the CCEs. The subject teacher is at liberty to decide which units are to be assessed using CCEs and which unit is to be assessed on the basis of IAT.The overall weightage of Continuous Internal Evaluation (CCE + IAT) shall be 40% of the total marks for the course. The remaining 60% of the marks shall be allocated to the semester-end examinations.The subject teachers are required to communicate the chosen CCE methods and the corresponding syllabus units to the students at the beginning of the semester to ensure clarity and proper preparation.

Practical Examination Credit 2: Pattern (30+20)

External Practical Examination (30 marks):

- Practical examination shall be conducted by the respective department at the end of the semester.
- Practical examination will be of 3 hours duration and shall be conducted as per schedule.
- Practical examination shall be conducted for 2 consecutive days for 2 hr/ day where incubation conditionis required.
- There shall be 05 marks for journal and viva-voce. Certified journal is compulsory to appear for practical examination.

Internal Practical Examination (20 marks):

- Internal practical examination of 10 marks will be conducted by department as per schedule given.
- For internal practical examination student must produce the laboratory journal of practicals completed along with the completion certificate signed by the concerned teacher and the Head of the department.
- There shall be continuous assessment of 30 marks based on student performance throughout the semester. This assessment can include quizzes, group discussions, presentations and other activities assigned by the faculty during regular practicals. For details refer internal theory examination guidelines.
- Finally 40 (10+30) marks performance of student will be converted into 20 marks.

SEMESTER-III

S.Y. B.Sc. Computer Science (Major)
Semester-III
CS-DSC-231: Data Structures-I

Total Hours: 30 **Credits: 2**

Course Objectives	• To familiar with linear data structures. • To familiar with basic techniques of algorithm analysis • To familiar with recursion techniques and its applications	
Course Outcomes	After successful completion of this course, students are expected to: • Apply and implement learned algorithms data structures to solve problems. • Meet the desired programming needs. • Apply searching techniques on data	
Unit	**Content**	**Hours**
Unit I	**Introduction to Data Structure, Algorithmic Notations and Algorithm Analysis** • Introduction to Data Structure • Types of data structure 1. Primitive 2. Non Primitive 3.Linear 4. Non linear Need of data structure • Algorithm Notations.:- Format Convention, Name of Algorithm, Introductory Comment, Steps, • Comments Data Structure:- Arrays,Dynamic Storage • allocation, Functions,Procedures • Rate of Growth, Basic time analysis of an algorithm, Order Notation , More timing Analysis, Space analysis of an algorithm	**8**
Unit II	**Static Data Strucutres : Stacks and Queues** • **Stacks** • Definition and concept • Representations • Operations – push, pop, peep, change • Applications – infix to postfix & prefix, postfix evaluation • Recursion • **Queues** • Definition and Concept, Representation – static, • Operations- Insert, Delete • Circular queue - Concept, Operations – insert, delete • DeQue – Concept, types, Operations- insert,delete • Priority queues – Concept	**8**
Unit III	**Dynamic Data Structures : Linked List** • Introduction to Linked list, Implementation of List – Dynamic representation. • Types of Linked List Singly Linked list : Operations- Insert, delete, search Circular linked list : Operations- Insert, delete, search Doubly linked linear list : Operations- Insert, delete, search	**10**

	• Applications of linked list – polynomial manipulation	
Unit IV	**Searching Techniques** • Linear Search • Binary Search • Hash Table Method: Introduction, Hashing Function,Collision Resolution Technique	**4**
Study Resources	• Trembley, J. P. and Soresan, P.G. (1983), An introduction to data structures with applications, Mc-Graw Hill International Editions, ISBN-13: 978-0070651579,ISBN-10: 0070651574 • Horowitz, E., and Sahani, S. (1973), Data Structures :Galgotia publication • Aho, Hopcroft, Ulman J.V. (1983), Data Structures and Algorithms, ISBN-13: 978-0201000238 ,ISBN-10: 0201000237 • Nikaulus, W. (1976) Algorithms- Data Structures Programs, ISBN-13: 978-130224187, ISBN-10: 0130224189 • Tannenbaum, A. M. (1995), Data Structures using C and C++; PHI., ISBN-13: 978-0130369970,ISBN-10: 0130369977	

S.Y. B.Sc. Computer Science (Major)
Semester-III
CS-DSC-232: Java Programming-I

Total Hours: 30 **Credits: 2**

Course Objectives	• To understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc. • To understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc. • To have the ability to write a computer program to solve specified problems. • To be able to use the Java SDK environment to create, debug and run simple Java programs.	
Course Outcomes	After successful completion of this course, students are expected to: • Identify classes, objects, members of a class and relationships among them needed for a specific problem • Write Java application programs using OOP principles and proper program structuring • Demonstrate the concepts of polymorphism and inheritance • Write Java programs to implement error handling techniques using exception handling	
Unit	**Content**	**Hours**
Unit I	**Introduction** • Introduction • History of Java • Comparison of Java and C++ • Features of Java • Java and Internet • JDK Environment (Java, Javac, Applet Viewer, Javadoc) • Simple Java Program • Java Program Structure • Implementing Java Program • Java Virtual Machine • Command Line Arguments	**7**
Unit II	**Classes and Objects and Methods** • Classes and Object • Defining a class • Adding variables ,methods • Creating objects • Accessing data members • Constructors • Method Overloading • Packages • Visibility Control • Inheritance-Extending a class • Overriding methods • Abstract methods and Classes • Reflection - 'Class'class	**8**

Unit III	**Array and Strings** • One –Dimensional Array • Creating array • Two- Dimensional Array • Strings • String functions-concatenation, Sub string, String editing, testing for equality, • Character extraction functions-CharAt,getChars,getByte, • Formatting functions, • Date and Time functions using Gregorian calendar class • Wrapper classes • Interfaces • Inner classes • Multithreading	**7**
Unit IV	**Exception and File Handling** • Types of errors • Exceptions • Syntax of Exception Handling Code • Multiple Catch statements • Using finally statement • Throwing your own exception • String class and StringBuffer Class • Stream classes-Byte Stream classes , Character Stream Classes • Creation of files • Reading/Writing characters and bytes • Random Access files	**8**
Study Resources	• Cay's Horstmann and Gary Cornell. Core Java Volume -1 Fundamentals • E. Balaguruswamy (Tata Mc Graw Hill) Programming with Java – A primer • Herbert Schildt (TMH) The complete reference JAVA-2 Fifth Edition • Java 6 Programming Black Book	

S.Y. B.Sc. Computer Science (Major)
Semester-III
CS-DSC-233: Practical on Data Structures and Java Programming-I

Total Hours: 60 **Credits: 2**

Course Objectives	• To provide the knowledge of basic data structures and their implementations. • To understand importance of data structures in context of writing efficient programs. • To develop skills to apply appropriate data structures in problem solving. • To get basic practical knowledge about the basic object oriented concepts.	
Course Outcomes	After successful completion of this course, students are expected to: • Aanalyze and compare various linear and non-linear data structures • Code, debug and demonstrate the working nature of different types of data structures and their applications. • Implement, analyse, and evaluate the searching algorithms. • Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.	
Sr. No.	**Content**	**Hours**
1	Write a program to implement Stack operations : push, pop, Display	**4**
2	Write a program to implement Stack operations : peep, change, Display	**4**
3	Write a program to implement Linear Queue operations : Insert, Delete, Display	**4**
4	Write a program to implement Circular queue with its operations: Insert, Delete, Display	**4**
5	Write a program to implement singly linked list with operations. i) create ii) insert at beginning iii) delete element from beginning	**4**
6	Write a program to implement singly linked list with operations. i) create ii) insert at the end position iii) delete element from last position	**4**
7	Write a program to implement singly linked list with operations. i) create ii) insert at given position iii) delete element from given position	**4**
8	Implement i) Linear Search ii) Binary Search	**4**
9	Write a program in Java to create student information using array.	**4**
10	Write a program in Java to implement user defined package.	**4**
11	Write a program in Java to implement default & parameterized constructor.	**4**
12	Write a program in Java to demonstrate various operations on string functions.	**4**

13	Write a program in Java to demonstrate wrapper classes	**4**
14	Write a program in Java to implement inheritance.	**4**
15	Write a program in Java to demonstrate exception handling.	**4**
Study Resources	• Trembley, J. P. and Soresan, P.G. (1983), An introduction to data structures with applications, Mc-Graw Hill International Editions, ISBN-13: 978-0070651579,ISBN-10: 0070651574 • Horowitz, E., and Sahani, S. (1973), Data Structures :Galgotia publication • Aho, Hopcroft, Ulman J.V. (1983), Data Structures and Algorithms, ISBN-13: 978-0201000238 ,ISBN-10: 0201000237 • Nikaulus, W. (1976) Algorithms- Data Structures Programs, ISBN-13: 978-130224187, ISBN-10: 0130224189 • Tannenbaum, A. M. (1995), Data Structures using C and C++; PHI., ISBN-13: 978-0130369970,ISBN-10: 0130369977	

***Mandatory to perform any 12 practical from above.**

S.Y. B.Sc. Computer Science (Major)
Semester-III
CS-SEC-231: HTML 5.0-I

Total Hours: 30 **Credits: 2**

Course Objectives	• To understand the basic structure of the Internet, web page and website • To be able to build a website. • To create and validate a web page. • To publish a web page.	
Course Outcomes	After successful completion of this course, students are expected to: • Learn to read, write and identify HTML5 tags in a page • Understand the basic structure of a web page • Understand directory structures and how they impact html code • Learn about CSS usage.	
Unit	**Contents**	**Hours**
Unit I	**Introduction of HTML 5.0** • What is HTML5? • History of HTML5 • Vision philosophy behind HTML5 • Getting Started with HTML5. • Working with Text • Working with Lists • Tables • Working with Hyperlinks • Images	**7**
Unit II	**Forms and Multimedia Controls in HTML 5** • Forms: need of Web Applications, • Current solutions, • New Input Types, • New Attributes, • Form Validation and Browser Support. • Audio and video: State of Web Audio and video based Plug-in, • State of Audio and video Codec, • New Audio/Video Mark-up, Attributes and method, • understanding Audio/video Events, • Customizing Audio/video controls	**8**
Unit III	**HTML Canvas** • Overview of graphics in the browser, • Using a canvas • Context and coordinates • Drawing shapes • working with paths • Drawing text and images • Working with pixels • understanding transforms • Browser support.	**7**

Unit IV	**CSS (Cascading Style Sheet)**
	• **CSS Introduction:**
	  • Id and class Selector
	  • CSS Styling: Styling backgrounds
	  • Styling text, styling fonts
	  • Styling links, styling lists
	  • Styling tables
	• **Graphics using CSS:**
	  • Box Model, Border, Outline
	  • Margin, Padding
	  • Advanced, Grouping/Nesting
	  • Dimension Display
	  • Positioning, floating
	  • Align, Navigation Bar
	  • Image gallery
	  • Image Opacity
	  • Image Sprites
	  • Hover animation
		8

Study Resources	• Pilgrim, M. (2009) HTML5 up and running O'reilly • Gauchat, J.D. (2012) HTML5 for Masterminds • Apress, Manian D. (2010), Beginning HTML5 and CSS3 • Frain B. (2011), Responsive web design with HTML5 and CSS3(2nd eddtion)	

S.Y. B.Sc. Computer Science (Major)
Semester-III
CS-CEP-231: Community Engagment Program

Total Hours: 60 **Credits: 2**

In alignment with the National Education Policy (NEP) 2020, MooljiJaitha College (Autonomous), Jalgaon is introducing the Community Engagement Program at the undergraduate level. The NEP 2020 emphasizes holistic development, inclusivity, and integrating vocational education with academic learning, aiming to nurture socially responsible individuals. Inspired by NEP 2020, the Community Engagement Program aim to produce knowledgeable, compassionate, and proactive graduates, contributing to a more just, equitable, and sustainable society. This course fosters a strong connection between education and socioeconomic problems of real-world. Students will learn about the challenges faced by vulnerable households and appreciate local wisdom and lifestyles.

Objectives

- To engage students in activities that promote emotional, social, and intellectual growth, fostering a well-rounded approach to personal and academic development.
- To provide hands-on experiences that complement classroom learning, enabling students to apply their knowledge in socioeconomic problems of real-world.
- To instil a sense of responsibility towards the community by encouraging students to actively participate in social and environmental initiatives, appreciate rural culture, lifestyle, and wisdom.

Outcomes

After completing this course, students will be able to

- Understand rural and/or urban culture, ethos, and socioeconomic realities.
- Develop a sense of empathy with the local community while appreciating the significant contributions of local communities to society and the economy.
- Learn to value the local community wisdom and identify opportunities for contributing to the community's socioeconomic improvements.

Activities

- Conduct workshops and interactive sessions on emotional intelligence and social skills.
- Organize debates, discussions, and intellectual challenges that stimulate critical thinking and socioeconomic problem-solving using concern subject.
- Organize field visits where students can work on real-world problems, such as environmental conservation, rural and/or urban planning, or community health.
- Organize internships or service-learning opportunities with local businesses, NGOs, or government agencies.
- Facilitate project-based learning activities that require students to use their academic knowledge to develop solutions to c`ommunity issues.
- Engage students in community service activities that address local social and environmental issues.
- Organize cultural exchange programs or field trips to rural areas to foster an appreciation of rural culture and wisdom.
- Facilitate collaborative projects involving students, educators, and community members to develop solutions for local challenges, promoting teamwork and collective problem-solving.

- Conduct educational sessions on the status of various agricultural and development programs and the challenges faced by vulnerable households, ensuring inclusivity and accessibility for all students.

S. No.	Module Title	Module Content	Assignment submission	Teaching/ Learning Methodology
1	**Appreciation of Rural Society**	Rural lifestyle, rural society, caste and gender relations, rural values with respect to community, nature and resources, elaboration of "soul of India lies in villages', rural infrastructure.	Prepare a map (physical, visual or digital) of the village you visited and write an essay about inter-family relations in that village.	– Classroom discussions – Field visit – Assignment
2	**Understanding rural and local economy and livelihood**	Agriculture, farming, land ownership, water management, animal husbandry, non-farm livelihoods and artisans, rural entrepreneurs, rural markets, migrant labour.	Describe your analysis of the rural house hold economy, its challenges and possible pathways to address. Circular economy and migration patterns.	– Field visit – Group discussions in class – Assignment
3	**Rural and local Institutions**	Traditional rural and community organisations, Self-help Groups, Panchayati raj institutions (Gram Sabha, Gram Panchayat, Standing Committees), Nagarpalikas and municipalities, local civil society, local administration.	How effectively are Panchayati Raj and Urban Local Bodies (ULBs) institutions functioning in the village? What would you suggest to improve their effectiveness? Present a case study (written or audio-visual).	– Classroom – Field visit – Group presentation of assignment
4	**Rural and National Development Programmes**	History of rural development and current national programmes in India: SarvaShikshaAbhiyan, BetiBachao, BetiPadhao, Ayushman Bharat, Swachh Bharat, PM AwaasYojana, Skill India, Gram Panchayat Decentralised Planning, National Rural Livelihood Mission (NRLM), Mahatma Gandhi National	Describe the benefits received and challenges faced in the delivery of one of these programmes in the local community; give suggestions about improving the implementation of the programme for the poor. Special focus on the urban informal sector and migrant households.	– Classroom – Each student selects one program for field visit – Written assignment

	Rural Employment Guarantee Act 2005 (MGNREGA), SHRAM, Jal Jeevan Mission, Scheme of Fund for Regeneration of Traditional Industries (SFURTI), AtmaNirbhar Bharat, etc.

Note: The modules are suggestive in nature and students can opt any one activities for community engagement program and field project based on topic appropriate to their regional community context.

Some additional suggestive themes for field-based / community engagement activities are listed below:

- Management curriculum may include aspects of micro-financing in a rural context;
- Chemistry syllabus can have a component of conducting water and soil analysis in surrounding field areas;
- Political science syllabus could include a mapping of local rural governance institutions and their functioning.
- Environment education will include areas such as climate change, pollution, waste management, sanitation, conservation of biological diversity, management of biological resources and biodiversity, forest and wildlife conservation, and sustainable development and living
- Understanding panchayats and constitutional mandate of local governance
- Panchayat administration, Gram Sabha, Mahila Sabha, Gram Panchayat Development Plan (GPDP), local planning of basic services.
- Micro-finance, SHGs, system of savings and credit for local business, linkages to banks, financial inclusion.
- Rural – entrepreneurship, opportunities for small business in local communities, access to financial and technical inputs to new entrepreneurs.
- Renewable energy, access to household and community level solar and bio-mass systems for sustainable energy use.
- Participatory Monitoring and evaluation of socio-economic development programmes, and cost-benefit analysis of project proposals.
- Migrant workers' livelihood security and social services.
- Hygiene and sanitation, improving health and personal behaviours, locally manageable decentralised systems and awareness against stubble burning.
- Water conservation, traditional practices of storage and harvesting, new systems of distribution and maintenance.
- Women's empowerment, gender inequality at home, community and public spaces, safety of girls and women, access to skills, credit and work opportunities.
- Child security, safety and good parenting, nutrition and health, learning and training for child care.
- Rural Marketing, market research, designing opportunities for rural artisans and crafts, and new products based on demand assessment.

- Community Based Research in Rural Settings, undertaking research that values local knowledge, systematises local practices and tools for replication and scale-up.
- Peri-urban development of informal settlements, mapping and enumeration, design of local solutions.

Assessment:

- Readings from related literature including e-content and reflections from field visits should be maintained by each student in the form of Field Diary (20 Marks)
- Submission of assignments based on modules assignment submission (details mentioned above) (20 Marks)
- Oral/ Group discussion/ Presentation (10 Marks)

S.Y. B.Sc. Computer Science (Minor)
Semester-III
CS-MIN-231: Programming in C-I

Total Hours: 30 **Credits: 2**

Course Objectives	• To study the fundamental programming languages • To know the process logic development • To give the practical knowledge of programming. • To be able to develop logics to create applications in C.	
Course Outcomes	**After successful completion of this course, students are expected to** • Enter basic logic development. • Prepare simple program • Prepare basic programming logic and entries • Obtain basic knowledge of program	
Unit	**Content**	**Hours**
Unit I	**Introduction** • Types of Programming languages, • History, features and application, • Simple program logic • program development cycle, • pseudocode statements and flowchart symbols • sentinel value to end a program • programming and user environments • evolution of programming models • desirable program characteristics.	**7**
Unit II	**Basics of C Language** • Overview of C: History of C, Importance of C, • Structure of a C Program. • Elements of C: C character set, identifiers and keywords, • Data types, Constants and Variables, • Assignment statement, Symbolic constant. • Input/output: Unformatted & formatted I/O • Input functions -scanf(), getch(), getche(), getchar(), gets(), • Output functions - printf(), putch(), putchar(), puts().	**8**
Unit III	**Control Flow and Logical Expressions** • Operators & Expression: Arithmetic operators • Relational, logical, bitwise, unary, assignment operators, • Conditional operators and special operators, operator hierarchy & associability • Decision making & branching: Decision making with IF statement, IF-ELSE statement, Nested IF statement, ELSE-IF ladder, switch statement, goto statement. • Loops control structure: while loop, for loop, do–while loop, nested loop, break , continue, switch, go to, exit statement	**8**
Unit IV	**Arrays and String** • Array Declaration and initialization • Array Manipulation	**7**

	• Multidimensional array • Dynamic Arrays • Strings – Declaration and initialization of string variables, Standard library string function strlen(), strcpy(), strcat(), strcmp(), arithmetic operations on characters	
Study Resources	• E. Balagurusamy, "Programming in ANSI C", 4/e, (TMH) • B. Kernighan & Dennis Ritchie, "The C Programming Language", 2/e PHI • Paul Deitel, Harvey Deitel, "C: How to Program", 8/e, Prentice Hall. • P.C. Sethi, P.K. Behera, "Programming using C", Kalyani Publisher, Ludhiana • Kernighan, B. W., & Ritchie, D. M. (1988). The C Programming Language (2nd ed.). Prentice Hall. • Prata, S. (2013). C Primer Plus (6th ed.). Addison-Wesley. • Deitel, P. J., &Deitel, H. M. (2015). C: How to Program (8th ed.). Pearson. • King, K. N. (2012). C Programming: A Modern Approach (2nd ed.). W. W. Norton & Company. • Gottfried, B. (2013). Programming in C (4th ed.). McGraw-Hill Education. • Hanly, J. J., &Koffman, E. B. (2011). Problem Solving and Program Design in C (7th ed.). Pearson. • Venugopal, K. N., & Prasad, P. R. (2012). Mastering C (2nd ed.). McGraw-Hill Education	

S.Y. B.Sc. Computer Science (Minor)
Semester-III
CS-MIN-232: Word Processig Tools

Total Hours: 30 **Credits: 2**

Course Objectives	• To create new document and save the data • To format the text in the document using different formatting tools • To arrange the text into columns and insert different objects into the text document. • To use mail merge on common document.	
Course Outcomes	After successful completion of this course, students are expected to: • Aware different word processing tools and have hands on one of them. • Learn different formatting options for text and document. • Embed the object into text document • Create document with effective graphics and formatting	
Unit	**Content**	**Hours**
Unit I	**Launch any Word processing tool and navigate the editing screen.** • Launch Word. • Identify the components of the Word window. • Edit a document. • Save a document. • Preview and print a document. • Close a document. • Locate and open an existing document. • Create a new document. • Close the Word application. • Use Print Preview. • Print a document	**7**
Unit II	**Create, edit a Text document and Apply formatting** • Create a letter. • Select text. • Cut, copy, and paste text. • Find and replace text. • Change fonts and font sizes. • Apply font styles, character styles, and special character effects. • Change the case of text. • Highlight text in a document. • Insert symbols and special characters. • Add bullets, numbering, borders, and shading. • Set line and paragraph spacing. • Align and indent paragraphs. • Insert page breaks. • Create and modify headers and footers. • Apply paragraph styles. • Create outlines.	**8**

Unit III	**Work** with columns, pictures, diagrams, and charts and tables. • Create and use newspaper columns. • Insert pictures. • Create diagrams. • Create and modify a data chart • Create basic tables. • Create and format tables. • Modify tables. • Customize tables. • Use special table features. • Use table styles. • Use tab stops in a table. • Convert text to a table. • Merge and split table cells.	**7**
Unit IV	**Use the mail merge wizard, Work with drawing objects and graphics** • Explain the steps of the mail merge process. • Define the main document. Select the data source. Merge the main • document and data source. • Create drawing objects. • Add decorative page borders. • Use WordArt special text effects. • Insert, position, and delete pictures. • Insert and edit text boxes. • Create and edit an organization chart. • Create an equation.	**8**
Study Resources	• Frandsen T., (2010), Microsoft Office Word 2007, BookBoon • Curtis F.(2007), Microsoft Office Word 2007 Step by Step • https://testbook.com/computer-awareness/microsoft-office • https://documentation.libreoffice.org/assets/Uploads/Documentation/en/GS5.0/PDF/GS5001-IntroducingLibreOffice.pdf • https://www.teachmint.com/tfile/studymaterial/b-com/ccc/libreofficecalcpdf/e66b485b-3992-44c9-8972-209158129d10	

S.Y. B.Sc. Computer Science (Minor)
Semester-III
CS-MIN-233: Practical on Programming in C-I

Total Hours: 60 **Credits: 2**

Course Objectives	• To study the fundamental programming languages • To know the process logic development • To give the practical knowledge of programming. • To be able to develop logics to create applications in C.	
Course Outcomes	**After successful completion of this course, students are expected to** • Enter basic logic development. • Prepare simple program • Prepare basic programming logic and entries • Obtain basic knowledge of program	
Sr. No.	**Content**	**Hours**
1	Write a program to declare some variables of type int, float and double. Assign some values to these variables and display these values.	4
2	Write a program to find the area and volume of rectangle, square, triangle andcircle.	4
3	Write a program to enter a number from the user and display the month name. If number >13 then display invalid input usingswitch case.	4
4	Write a program to check whether the number is even orodd.	4
5	Write a program to check whether the number is positive, negativeor zero.	4
6	Write a program to check whether the entered number is prime ornot	4
7	Write a program to find the largest of threenumbers.	4
8	Write a program to print the Fibonacciseries.	4
9	Write a program to find whether a given number is palindrome ornot	4
10	Write a program to check whether the entered number isArmstrong or not.	4
11	Write a program to find the largest value that is stored in thearray	4
12	Write a program using pointers to compute the sum of allelements stored in anarray.	4
13	Write a program that performs multiplication ofmatrices.	4
14	Write a program to demonstrate all sting functions.	4
15	Write a program to read and display information of student including Rollno, Name, Address, Marks of three subject and display total marks	4
Study Resources	• E. Balagurusamy, “Programming in ANSI C”, 4/e, (TMH) • B. Kernighan & Dennis Ritchie, “The C Programming Language”, 2/e PHI • Paul Deitel, Harvey Deitel, “C: How to Program”, 8/e, Prentice Hall. • P.C. Sethi, P.K. Behera, “Programming using C”, Kalyani Publisher, Ludhiana • Kernighan, B. W., & Ritchie, D. M. (1988). The C Programming Language (2nd ed.). Prentice Hall. • Prata, S. (2013). C Primer Plus (6th ed.). Addison-Wesley.	

***Mandatory to perform any 12 practical from above.**

S.Y. B.Sc. Computer Science (Open Elective)
Semester-III
CS-OE-231: Professional Presentation Skills

Total Hours: 30 **Credits: 2**

Course Objectives	**Student will be able to** • customise the formatting of charts presentation • deal with shapes and images • use media in presenation • create and edit a PivotChart	
Course Outcomes	**After successful completion of this course students are expected to :** • Demonstrating the basic mechanics and navigation of an spreadsheet. • Learning the use and utility of functions and formulas on spreadsheet. • Using clip art to enhance ideas and information in Excel worksheets. • Analyzing data using Pivot Tables and Pivot Charts.	
Unit	**Content**	**Hours**
Unit I	**Create and Manage Presentations** • Create a Presentation • Insert and Format Slides • Modify Slides, Handouts, and Notes • Change Presentation Options and Views • Configure a Presentation for Print • Configure and Present a Slide Show.	**7**
Unit II	**Insert and Format Text, Shapes, and Images** • Insert and Format Text • Insert and Format Shapes and Text Boxes • Insert and Format Images • Order and Group Objects	**8**
Unit III	**Insert Tables, Charts, SmartArt, and Media** • Insert and Format Tables • Insert and Format Charts • Insert and Format SmartArt graphics • Insert and Manage Media	**7**
Unit IV	**Apply Transitions and Animations** • Apply Slide Transitions • Animate Slide Content • Set Timing for Transitions and Animations	**8**
Study Resources	• Frandsen T., (2010), Microsoft Powerpoint 2007, BookBoon • Curtis F.(2007), Microsoft Office Powerpoint Step by Step • https://testbook.com/computer-awareness/microsoft-office • https://documentation.libreoffice.org/assets/Uploads/Documentation/en/GS5.0/PDF/GS5001-IntroducingLibreOffice.pdf • https://www.teachmint.com/tfile/studymaterial/b-com/ccc/libreofficecalcpdf/e66b485b-3992-44c9-8972-209158129d10	

SEMESTER-IV

S.Y. B.Sc. Computer Science (Major)
Semester-IV
CS-DSC-241: Data Structures-II

Total Hours: 30 **Credits: 2**

Course Objectives	• To be familiar with non-linear data structures. • To be familiar with algorithm analysis of different algorithms. • To be familiar with advanced data structures such as AVL trees, B-trees.	
Course Outcomes	After successful completion of this course, students are expected to: • Apply and implement learned algorithms data structures to solve problems. • Meet the desired programming needs.	
Unit	**Content**	**Hours**
Unit I	**Tree** • Definition and Concept, Binary tree, Storage representation and Manipulation of Binary trees • Sequential Storage representation of Binary Tree, Linked Storage representation of Binary Tree • Threaded storage representation of Binary Tree, Operations on Binary tree – Traversing • Operations & Algorithms on BST – Create, Insert, Delete • Search Trees- AVL Tree, single and double rotations, B-Trees- insertion and deletion, • Introduction to B+ and B* Trees	**7**
Unit II	**Graph** • Definition and Concept, Matrix representation of graph, • List Structures , Multi list representation of Graph • Traversal of graph : Breadth First Search • Depth First search • Applications of graph	**8**
Unit III	**Sorting** • Introduction • Sorting Techniques : Selection Sort • Insertion sort • Bubble Sort • Merge Sort • Heap Sort • Quick Sort • Radix Sort • Sorting Method Comparison on Time and space Complexity attribute	**7**
Unit IV	**File Structure** • Introduction to file • Sequential File concept • Index Sequential File concept • Direct file concept	**8**

Study Resources	• Trembley, J. P. and Soresan, P.G. (1983), An introduction to data structures with applications, Mc-Graw Hill International Editions, ISBN-13: 978-0070651579,ISBN-10: 0070651574 • Horowitz, E., and Sahani, S. (1973), Data Structures :Galgotia publication • Aho, Hopcroft, Ulman J.V. (1983), Data Structures and Algorithms, ISBN-13: 978-0201000238 ,ISBN-10: 0201000237 • Nikaulus, W. (1976) Algorithms- Data Structures Programs, ISBN-13: 978-130224187, ISBN-10: 0130224189 • Tannenbaum, A. M. (1995), Data Structures using C and C++; PHI., ISBN-13: 978-0130369970,ISBN-10: 0130369977	

S.Y. B.Sc. Computer Science (Major)
Semester-IV
CS-DSC-242: Java Programming-II

Total Hours: 30 **Credits: 2**

Course Objectives	• To understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc. • To understand fundamentals of object-oriented programming in java, including defining classes, invoking methods, using class libraries, etc. • To have the ability to write a computer program to solve specified problems. • To be familiar with GUI programming.	
Course Outcomes	**After successful completion of this course, students are expected to:** • Write Java programs to implement GUI programming. • Deal with event handling. • Deal with swing components • Deal with web pages using applets.	
Unit	**Content**	**Hours**
Unit I	**Unit 1: Graphics Programming** • Introduction-frames • frame Layouts • Displaying information in a frame • Graphics objects and paint component method • Text and fontsColors • Drawing shapes • Filling shapes • Paint mode and images	**7**
Unit II	**Event Handling** • Basic Event Handling • The AWT event hierarchy • Event handling summary-event sources and listener • Aadapter classes • Focus event • Window event • Keyboard event • Mouse events • Multicasting	**8**
Unit III	**Swing** • Introduction to layout management - Panels, Border Layout, GridLayout • Basics of Swing • Text Input- Text Field, Text Area, Passwordfield • Labels and Buttons • Making choices - Check boxes, Radio buttons, List, Comboboxes • JProgressBar class • JSlider class • JTable Class	**7**

	• Graphics in swing • Displaying Image • Menus-Building menus • Menu events • Popup menu • Keyboard mnemonics and Accelerators • Enabling and disabling menus • Dialog boxes-opening dialogs using inbuilt dialog box	
Unit IV	**Applets** • Introduction to applet • How applet differ from Applications • Converting application to applets • Life cycle of applet • Creating Executable Applet • Applet Tag • Param Tag	**8**
Study Resources	• Cay's Horstmann and Gary Cornell. Core Java Volume -1 Fundamentals • E. Balaguruswamy (Tata Mc Graw Hill) Programming with Java – A primer • Herbert Schildt (TMH) The complete reference JAVA-2 Fifth Edition • Java 6 Programming Black Book	

S.Y. B.Sc. Computer Science (Major)
Semester-IV
CS-DSC-243: Practical on Data Structures and Java Programming-II

Total Hours: 60 **Credits: 2**

Course Objectives	• To be familiar with non-linear data structures. • To be familiar with algorithm analysis of different algorithms. • To be familiar with advanced data structures such as AVL trees, B-trees. • To get basic practical knowledge about the Swing Components. • To acquaint concepts of Applets.	
Course Outcomes	After successful completion of this course, students are expected to: • Apply and implement learned algorithms data structures to solve problems. • Meet the desired programming needs. • Use various swing components of java. • Execute Applet Programming	
Sr. No.	**Content**	**Hours**
1	To Create a binary tree and Implement Inorder Traversal Techniques.	**4**
2	To Create a binary tree and Implement Preorder Traversal Technique.	**4**
3	To Create a binary tree and Implement Postorder Traversal Technique.	**4**
4	Implement BFS Graph Search Technique.	**4**
5	Implement DFS Graph Search Technique.	**4**
6	Implement Bubble sort technique.	**4**
7	Implement Selection sort technique.	**4**
8	Implement Insertion sort technique.	**4**
9	Implement Merge sort technique.	**4**
10	Implement Quick sort technique.	**4**
11	Write a program in Java to display messages in various fonts in a frame	**4**
12	Write a program in Java to demonstrate window events.	**4**
13	Write a program in Java to demonstrate Keyboard events.(key pressed, key released)	**4**
14	Write a program in Java to demonstrate Mouse events.	**4**
15	Write a program in Java to demonstrate multicasting	**4**
16	Write an Applet to display human face.	**4**
Study Resources	• Trembley, J. P. and Soresan, P.G. (1983), An introduction to data structures with applications, Mc-Graw Hill International Editions, ISBN-13: 978-0070651579,ISBN-10: 0070651574	

	• Horowitz, E., and Sahani, S. (1973), Data Structures :Galgotia publication • Aho, Hopcroft, Ulman J.V. (1983), Data Structures and Algorithms, ISBN-13: 978-0201000238 ,ISBN-10: 0201000237 • Nikaulus, W. (1976) Algorithms- Data Structures Programs, ISBN-13: 978-130224187, ISBN-10: 0130224189 • Tannenbaum, A. M. (1995), Data Structures using C and C++; PHI., ISBN-13: 978-0130369970,ISBN-10: 0130369977	

***Mandatory to perform any 12 practical from above.**

S.Y. B.Sc. Computer Science (Major)
Semester-IV
CS-SEC-241: HTML 5.0- II

Total Hours: 30 **Credits: 2**

Course Objectives	▪ To understand the basic structure of the Internet, web page and website ▪ To be able to build a website. ▪ To create and validate a web page. ▪ To publish a web page.	
Course Outcomes	After successful completion of this course, students are expected to: ▪ Learn to read, write and identify HTML5 tags in a page ▪ Understand the basic structure of a web page ▪ Understand directory structures and how they impact html code ▪ Learn about CSS usage.	
Unit	**Contents**	**Hours**
Unit I	**Selectors and Pseudo classes** • Attributes Selectors • The Target Pseudo-class • UI Element States Pseudo-classes • Negation Pseudo-class • Structural Pseudo-classes,	**7**
Unit II	**Fonts and text Effects** • Font on the web • Font services • @Font-face Rule • Text shadow • Word Wrapping Colors • Gradients • Background Images and Masks: Color • The opacity Property • Backgrounds • Background-origin and background-size.	**8**
Unit III	**Border and box effects** • Images borders • Rounded corners • Box shadow Transitions • Transforms and Animations: Transitions and transforms • Layouts:Colums and Flexible Box Layout • Flexible Box Model • Vendor Prefixes:What are Vendor prefixes? Strategies • Embedding Media: Video Formats, Styling video.	**7**
Unit IV	**Java Scripts** • Introduction to JavaScript • JavaScript Basics- Data Types • Control Structure	**8**

	• JavaScript Functions • Working with events • JS Popup boxes • JavaScript Objects • JavaScript HTML DOM	
Study Resources	• Pilgrim, M. (2009) HTML5 up and running O'reilly • Gauchat, J.D. (2012) HTML5 for Masterminds • Apress, Manian D. (2010), Beginning HTML5 and CSS3Frain B. (2011) • Responsive web design with HTML5 and CSS3(2nd eddtion)	

S.Y. B.Sc. Computer Science (Major)
Semester-IV
CS-SEC-242: Practical on HTML 5.0

Total Hours: 60 **Credits: 2**

Course Objectives	▪ To understand the basic structure of the Internet, web page and website ▪ To be able to build a website. ▪ To create and validate a web page. ▪ To publish a web page.	
Course Outcomes	After successful completion of this course, students are expected to: ▪ Learn to read, write and identify HTML5 tags in a page ▪ Understand the basic structure of a web page ▪ Understand directory structures and how they impact html code ▪ Learn about CSS usage.	
Sr. No.	**Contents**	**Hours**
1	Create a web page using HTML to display resume using text formatting tags, image, list, table, hyperlink.	**4**
2	Create a web page using HTML to create college admission form using validation, date and input types.	**4**
3	Create a web page using HTML to demonstrate audio and video tags.	**4**
4	Create a web page using HTML to create animated images using marquee tag.	**4**
5	Create a web page using HTML to create canvas which includes shapes, path, text and images.	**4**
6	Create a web page to demonstrate CSS for styling background, text, font, link, list and table.	**4**
7	Create a web page to demonstrate use of nesting in CSS.	**4**
8	Create a web page to display top navigation bar using CSS.	**4**
9	Create a web page to display various font effect, text shadow and text word wrapping using CSS.	**4**
10	Create a web page to demonstrate various gradients and backgrounds in CSS. Also demonstrate opacity property.	**4**
11	Create a web page to demonstrate use of flexible box model in CSS.	**4**
12	Create a web page using CSS to demonstrate image shadow.	**4**
13	Create a web page to demonstrate various image border styles.	**4**
14	Create a web page to demonstrate CSS for change background image on button click.	**4**
15	Create a web page Using HTML, CSS, JavaScript create a hover effect where the gradient follows the mouse cursor.	**4**
Study Resources	▪ https://programmingtrick.com/html-assignments • Pilgrim, M. (2009) HTML5 up and running O'reilly • Gauchat, J.D. (2012) HTML5 for Masterminds • Apress, Manian D. (2010), Beginning HTML5 and CSS3Frain B. (2011) ▪ Responsive web design with HTML5 and CSS3(2nd eddtion)	

S.Y. B.Sc. Computer Science (Major)
Semester-IV
CS-FP-241: Field Project

Total Hours: 60 **Credits: 2**

Preamble

In alignment with the National Education Policy (NEP) 2020, Moolji Jaitha College (Autonomous), Jalgaon is introducing the Field Project at the undergraduate level. The NEP 2020 emphasizes holistic development, inclusivity, and integrating vocational education with academic learning, aiming to nurture socially responsible individuals. This course fosters a strong connection between education and real-world applications. These initiatives aim to bridge the gap between theoretical knowledge and practical experience, helping students develop critical thinking, problem-solving skills, and a sense of civic responsibility.

Objectives

- To provide students with practical exposure in rural and urban socioeconomic context.
- To develop students abilities to apply subject knowledge to address real world problems
- To foster critical thinking and innovative approaches to solve socioeconomic issues.

Outcomes

After completing this course, students will be able to

- Participate actively in filed projects that benefit local communities and promote sustainable development practices.
- Analyse the socio economic data using appropriate methods showcasing improved problem-solving skills, technical proficiency.
- Demonstrate the ability to apply theoretical knowledge to real-world situations effectively and exhibit communication skills.

Course structure

The course is divided in to four probable phases

I] Orientation and preparation

- Introduce to the course, Objectives and expectation
- Overview of socioeconomic development issues in rural and urban context
- Training on working methodology and data collection techniques
- Review existing literature related to topic to understand the background and context.

II] Work plan and Field visit

- Visit the potential sites to get a sense of the environment and logistical requirements.
- Create a detailed project plan outlining the steps, timeline, resources needed, and roles of team members.
- Obtain necessary approvals (Ethical/ local authorities/organizations/communities)
- Gather materials and resources (recording devices, cameras, notebooks and supplies)
- Conduct Preliminary Survey, choose appropriate methods for data collection and analysis (e.g., surveys, interviews, observations).

III] Data collection and analysis

- Pilot test to identify issues with data collection.

- Collect data systematically, ensuring consistency and accuracy.
- Keep detailed records of all data (field notes, recordings, photographs etc)
- Organize and analyse the data (manual/ software)

IV] Interpretation and Reporting

- Interpret your findings in the context to Objectives.
- Write and submit a comprehensive report detailing your methodology, findings, analysis, and conclusions. (Include visuals - charts, graphs, and photographs).
- Prepare a presentation to share findings with peers/ instructors/ community.

Assessment

- Field work participation, field note book, team work etc. (10 Marks)
- Data Collection and Analysis (15 Marks)
- Field project report (15 Marks)
- Presentation of Findings (10 Marks)

Examples of activities to be conducted under field projects

- **Biodiversity Survey**: Conduct a biodiversity survey in a local park or nature reserve, documenting plant and animal species.
- **Water Quality Testing**: Test water samples from different sources (e.g., rivers, lakes, groundwater) for pollutants and compare results.
- **Soil Analysis**: Collect soil samples from various locations and analyse their composition and quality.
- **Wildlife Tracking**: Use camera traps or tracking devices to monitor and study the behaviour of local wildlife.
- **Urban Heat Island Effect**: Measure and map temperature differences in various parts of a city.
- **Land Use Mapping**: Create maps showing different land uses in a region and analyze changes over time.
- **Cultural Heritage Documentation**: Document and analyze local cultural heritage sites or practices.
- **Community Interviews**: Conduct interviews with community members to understand social dynamics and traditions.
- **Ethnographic Study**: Participate in and observe community events to gather ethnographic data.
- **Crop Yield Analysis**: Study the factors affecting crop yield in different fields or under different farming practices.
- **Pest Management**: Investigate the effectiveness of various pest management techniques in local farms.
- **Sustainable Farming Practices**: Evaluate the impact of sustainable farming practices on soil health and crop productivity.
- **Community Needs Assessment**: Conduct surveys and interviews to identify the needs and concerns of a community.
- **Social Network Analysis**: Study the social networks within a community to understand relationships and influence.
- **Public Health Study**: Investigate public health issues in a community, such as access to healthcare or prevalence of diseases.
- **Infrastructure Survey**: Assess the condition and effectiveness of local infrastructure, such as roads, bridges, and buildings.

- **Renewable Energy Potential**: Evaluate the potential for renewable energy sources (e.g., solar, wind) in a specific area.
- **Water Management**: Study and improve local water management systems, including irrigation and drainage.
- **Literacy Program Evaluation**: Evaluate the effectiveness of local literacy programs and suggest improvements.
- **Educational Resource Assessment**: Assess the availability and quality of educational resources in local schools.
- **Market Analysis**: Conduct a market analysis for a local business or industry.
- **Entrepreneurship Project**: Develop a business plan for a local entrepreneurial venture
- **Local History Documentation**: Research and document the history of a local site, building, or community.
- **Oral History Project**: Conduct interviews with local residents to collect oral histories and preserve community memories.
- **Archival Research**: Explore local archives to uncover historical documents and artifacts related to a specific topic or period.
- **Community Mural**: Design and create a mural in collaboration with community members that reflects local culture and history.
- **Public Art Installation**: Develop and install a public art project that engages the local community.
- **Art Exhibit Curation**: Curate an exhibit featuring works by local artists, highlighting themes relevant to the community.
- **Music Documentation**: Record and document traditional or contemporary music from the local area.
- **Community Concerts**: Organize and perform in community concerts that showcase local musical talent.
- **Community Theatre Production**: Develop and produce a play that involves community members as actors and crew.
- **Site-Specific Theatre**: Create a theatrical performance that takes place in a non-traditional venue, such as a historic site or public space.
- **Cultural Mapping**: Map cultural resources and heritage sites within the community and analyze their significance.
- **Festival Documentation**: Document and analyze local festivals or cultural events, exploring their history and impact.
- **Ethnographic Study**: Conduct an ethnographic study of a particular cultural practice or community group.
- **Public Philosophy Discussions**: Organize and facilitate public discussions on philosophical topics relevant to the community.
- **Community Documentary**: Create a documentary film about a local issue, event, or group.
- **Digital Storytelling**: Develop digital storytelling projects that capture and share local stories.
- **Language Survey**: Conduct a survey of languages spoken in the community and analyze patterns of language use and change.
- **Dialect Study**: Study and document local dialects or accents, exploring their features and origins.
- **Language Preservation**: Work with community members to document and preserve endangered languages or dialects.
- **Gentrification Impact Study**: Examine the effects of gentrification on local communities, including displacement and economic changes.
- **Crime and Safety Analysis**: Study crime patterns and perceptions of safety within a community.
- **Ritual and Festival Study**: Participate in and document local rituals or festivals to understand their social and cultural significance.

- **Migration Patterns Study**: Analyze migration patterns and their effects on both the sending and receiving communities.
- **Food and Culture Study**: Investigate the role of food in cultural practices and social interactions within a community.
- **Local Governance Analysis**: Study the structure and functioning of local government and its impact on the community.
- **Political Participation Study**: Analyze patterns of political participation and engagement within a community.
- **Public Policy Impact Assessment**: Evaluate the impact of specific public policies on local communities.
- **Election Study**: Analyze voting behavior and patterns in local elections.
- **Mental Health Survey**: Conduct surveys to assess the mental health needs and resources in a community.
- **Social Behavior Observation**: Observe and analyze social behaviors in public spaces, such as parks or markets.
- **Stress and Coping Study**: Investigate sources of stress and coping mechanisms within a community.
- **Community Support Systems**: Study the role and effectiveness of community support systems and networks.
- **Youth Development Programs**: Evaluate the impact of youth development programs on community wellbeing.
- **Educational Equity Study**: Assess disparities in educational resources and Outcomes in local schools.
- **Parent and Teacher Interviews**: Conduct interviews to understand perceptions of educational quality and challenges.
- **After-School Program Evaluation**: Evaluate the effectiveness of after-school programs in supporting student development.
- **Educational Attainment Study**: Analyze factors influencing educational attainment in a community.
- **Local Economy Analysis**: Study the structure and dynamics of the local economy, including key industries and employment patterns.
- **Small Business Survey**: Conduct surveys of local small businesses to understand their challenges and successes.
- **Economic Impact of Events**: Analyze the economic impact of local events or festivals on the community.
- **Income Inequality Study**: Investigate patterns and causes of income inequality within a community.
- **Housing Affordability Analysis**: Study housing affordability issues and their impact on residents.
- **Gender Roles and Expectations**: Study gender roles and expectations within a community and their impact on individuals.
- **Women's Health Study**: Investigate issues related to women's health and access to healthcare.
- **Gender-Based Violence Survey**: Conduct surveys to understand the prevalence and impact of gender-based violence.
- **Workplace Equality Study**: Analyze gender equality in local workplaces, including pay equity and job opportunities.
- **Urban Development Projects**: Study the impact of urban development projects on local communities.
- **Public Space Usage**: Analyze how public spaces are used and perceived by different community members.

- **Transportation Study**: Investigate transportation needs and challenges within a community.
- **Green Space Analysis**: Study the availability and usage of green spaces in urban areas and their impact on residents.

S.Y. B.Sc. Computer Science (Minor)
Semester-IV
CS-MIN-241: Programming in C-II

Total Hours: 30 **Credits: 2**

Course Objectives	• Understand the concept of a program (i.e., a computer following a series of instructions) • Understand the concept of a variable holding a value, how a variable is declared and how it can change • Understand the concept of a loop – that is, a series of statements which is written once but executed repeatedly- and how to use it in a programming language • Be able to use a conditional statement to select a choice from two or more alternatives • Be able to break a large problem into smaller parts, writing each part as a module or function	
Course Outcomes	**After successful completion of this course, students are expected to** • Control the sequence of the program and give logical outputs • Implement strings in your C program • Store different data types in the same memory • Manage I/O operations in your C program • Repeat the sequence of instructions and points for a memory location • Apply code reusability with functions and pointers • Understand the basics of file handling mechanisms	
Unit	**Content**	**Hours**
Unit I	**User Defined Functions** • Definition, Need of user defined functions • Function prototype, • Parameter Passing Mechanism • Scope of variable • Nesting Functions • Recursion • Passing arrays to function • Passing string to function	**7**
Unit II	**Pointers** • Definition and declaration • Uses, Initialization • address operator • pointer arithmetic • dynamic memory allocation • arrays and pointers • pointer to function	**8**
Unit III	**Structure, Union and File** • Structure : Defining • Declaring, Accessing • Initialization Structure • nested structure • self-referential structure, bit-field	**7**

	• Arrays of Structures • Structures and Functions • Union: Difference between structure and union • active data member, structure within union • Self-referential Structure.	
Unit IV	**File Management** • Defining and Opening a File • File opening modes (read, write, append) • Closing a File, File operations • File and stream, • Error Handling During I/O Operations • Sequential and random access file, low level and high level file.	**8**
Study Resources	• E. Balagurusamy, "Programming in ANSI C", 4/e, (TMH) • B. Kernighan & Dennis Ritchie, "The C Programming Language", 2/e PHI • Paul Deitel, Harvey Deitel, "C: How to Program", 8/e, Prentice Hall. • P.C. Sethi, P.K. Behera, "Programming using C", Kalyani Publisher, Ludhiana • Kernighan, B. W., & Ritchie, D. M. (1988). The C Programming Language (2nd ed.). Prentice Hall. • Prata, S. (2013). C Primer Plus (6th ed.). Addison-Wesley. • Deitel, P. J., &Deitel, H. M. (2015). C: How to Program (8th ed.). Pearson. • King, K. N. (2012). C Programming: A Modern Approach (2nd ed.). W. W. Norton & Company. • Gottfried, B. (2013). Programming in C (4th ed.). McGraw-Hill Education. • Hanly, J. J., &Koffman, E. B. (2011). Problem Solving and Program Design in C (7th ed.). Pearson. • Venugopal, K. N., & Prasad, P. R. (2012). Mastering C (2nd ed.). McGraw-Hill Education.	

S.Y. B.Sc. Computer Science (Minor)
Semester-IV
CS-MIN-242: Practical on Programming in C-II

Total Hours: 60 **Credits: 2**

Course Objectives	• . Understand the concept of a program (i.e., a computer following a series of instructions) • Understand the concept of a variable holding a value, how a variable is declared and how it can change • Understand the concept of a loop – that is, a series of statements which is written once but executed repeatedly- and how to use it in a programming language • Be able to use a conditional statement to select a choice from two or more alternatives • Be able to break a large problem into smaller parts, writing each part as a module or function	
Course Outcomes	**After successful completion of this course, students are expected to** • Control the sequence of the program and give logical outputs • Implement strings in your C program • Store different data types in the same memory • Manage I/O operations in your C program • Repeat the sequence of instructions and points for a memory location • Apply code reusability with functions and pointers • Understand the basics of file handling mechanisms	
	Content	**Hours**
1.	Write a program to demonstrate user defined function in C.	4
2.	Write a program in C to find the square of any number using the function.	4
3.	Write a program to swap two numbers using call by reference.	4
4.	Write a program to find the factorial of a number using recursive function	4
5.	Write a program to demonstrate the use of pointers.	4
6.	Write a program to perform addition and subtraction of two pointer variables.	4
7.	Write a C program to input and print array elements using pointer.	4
8.	Write a C program to reverse an array using pointers.	4
9.	Write a C program to sort array using pointers.	4
10.	Write a C program to read and print an employee's detail using structure.	4
11.	Write a C program to demonstrate example of nested structure.	4
12.	Write a C program to declare, initialize an union, example of union.	4
13.	Write a C program to create and store information in a text file.	4
14.	Write a C program to read an existing file.	4
15.	Write a C program to read the file and store the lines in an array.	4
Study Resources	• E. Balagurusamy, “Programming in ANSI C”, 4/e, (TMH) • B. Kernighan & Dennis Ritchie, “The C Programming Language”, 2/e PHI	

	• Paul Deitel, Harvey Deitel, "C: How to Program", 8/e, Prentice Hall. • P.C. Sethi, P.K. Behera, "Programming using C", Kalyani Publisher, Ludhiana • Kernighan, B. W., & Ritchie, D. M. (1988). The C Programming Language (2nd ed.). Prentice Hall. • Prata, S. (2013). C Primer Plus (6th ed.). Addison-Wesley.	

***Mandatory to perform any 12 practical from above.**

S.Y. B.Sc. Computer Science (Open Elective)
Semester-IV
CS-OE-241: Basics of Internet

Total Hours: 30 **Credits: 2**

Course Objectives	• To understand the concept of computer network. • To make the student aware of types of websites. • To understand the web design process • To understand the various web page types and its navigation.	
Course Outcomes	After successful completion of this course, students are expected to: • Aware of computer networks and internet concepts. • Differtiate between various site types and their organizational models. • Understand the web design process can use the approcach for website development. • Understand the various page types.	
Unit	**Content**	**Hours**
Unit I	**Programming Concepts of network** • What is Computer Network? • Types of Networks (with Features and Application): LAN, WAN, MAN Wired Network, Wireless Network, MANET, Internet • Study of Web Browsers • Google services • Search Engines • Computer Virus • Computer Ethics	**7**
Unit II	**Introduction to Website** • Site Types • Site Structure • Site Organization Model • Site Planning and Testing	**8**

Unit III	**Web Design Process**
	• What is Web Design?
	• Web Design Pyramid
	• Web Process Model
	  • Modified Waterfall Model
	  • Joint Application Development Model
		7

Unit IV	**Page Types and Navigation Theory** • Page Types • Page Size and Margins • Web Page Categorization: Splash Page, Entrance Page, Home Page, Sub Home Page, Content Page, Exit Page. • What is Navigation and types of Navigation?	**8**
Study Resources	• Powell T. (2003), The Complete reference –Web Design, 2nd Ed. McGraw-Hill Education,	

	• Corner D., (2006), The Internet Book: Everything You Need to Know about Computer Networking and How the Internet Works, 4th Ed. Pearson • Web reference: www.w3schools.com	