SEMESTER – IV

USMABn20 - R Programming Language

Year: II **SEM:** IV	**Course Code:** USMABn20	**Title of the Course:** R Programming Language	**Course Type:** Theory	**Course Category:** Skill Based Elective	**H/W** 2	**CREDITS** 2	**HOURS** 100

Course Objectives

1. To introduce students to the concept of basic R programming, thereby enhancing the logical thinking of the students with regard to programming.
2. To train the students to apply the programming concepts of R to statistical investigations and problem solving.

Course Learning Outcomes (CLO)

The learners will be able to

1. Familiarize the basics of programming in R such as vectors, arrays, data frames, etc.
2. Use the Decision making-branching and looping statements in R programming.
3. Represent data and Interpret results through graphical tools in R.
4. Calculate basic statistical measures and fit standard distributions using R.
5. Understand and apply the programming concepts of R to perform tests of significance.
6. Understand and apply the programming concepts of R to perform Analysis of Variance.

CO	**PSO**					
	PSO1	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**
CO1	H	H	H	H	L	H
CO2	H	M	H	M	L	H
CO3	H	M	H	M	L	H
CO4	H	H	H	H	L	H
CO5	H	M	H	H	L	H

CO	**PO**					
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
CO1	H	H	H	M	M	H
CO2	H	H	H	M	M	H
CO3	H	H	H	M	M	H
CO4	H	H	H	M	M	H
CO5	H	H	H	M	M	H

(L-Low, M-Moderate, H-High)

Course Syllabus

Unit I: Basics of R (6 hours)

Introduction and Preliminaries-Simple Manipulations; Numbers and Vectors-Arrays and Matrices - Lists and Data Frames-Reading Data from files (K1, K2, K3, K4)

Unit II: Decision Making and Graphical Procedures (6 hours)

Grouping-Loops and Conditional Execution-Graphics on R-Scatter Plot-Line Graphs-Pie Charts-Bar Plots-Histograms-Frequency Polygons (K1, K2, K3, K4)

Unit III: Statistical Measures & Probability Distributions (6 hours)

Mean, Median and Mode-Variance, Standard Deviation and Mean Deviation -Correlation and Regression-Standard Distributions -Binomial, Poisson and Normal Distributions (K1, K2, K3, K4)

Unit IV: Tests of significance (6 hours)

z-Test-Test for Mean-Test for Proportion-Comparing two Means-Comparing two proportions-Student t-test and t-test for two Means- Chi-Square Test-Test for Independence of Attributes (K1, K2, K3, K4)

Unit V: Analysis of Variance (6 hours)

Comparing more than two Means-Completely Randomized Design - One-Way Classification-Randomized Block Design-Two-Way Classification-Latin Square Design (K1, K2, K3, K4)

Text Books:

1. The R Book-Michael J. Crawley-Imperial College London at Silwood Park, UK, Second Edition, A John Wiley & Sons, Ltd., Publication, 2013.
2. An Introduction to R-Notes on R: A Programming Environment for Data Analysis and Graphics W. N. Venables, D. M. Smith and the R Core Team-(Version 3.6.3), 2020.

Reference Books:

1. The Art of R Programming A Tour of Statistical Software Design-Norman Matloff, No Starch Press, San Francisco, 2011.
2. Introduction to Statistics with R - Anne Segonds-Pichon, Babraham Bioinformatics, 2015.
3. R for Dummies, Andrie de Vries and JorisMeys, 2nd Edition, John Wiley & Sons, Inc., 2015.

e-Resources:

1. https://nptel.ac.in/
2. www.coursera.org
3. https://spoken-tutorial.org

SEMESTER – V

UEMAA20 – Programming in C

Year: III **SEM:** V	**Course Code:** UEMAA20	**Title of the Course:** Elective-I A: Programming in C	**Course Type:** Theory	**Course Category:** Core Elective	**H/W** 4	**CREDITS** 3	**HOURS** 100

Course Objectives

1. To introduce students to the concept of basic programming, thereby enhancing the logical thinking of the students with regard to programming.
2. To train the students to apply the programming concepts of C to mathematical investigations and problem solving.
3. To enhance the ability of students to work independently and do in-depth study of various notions of programming.

Course Outcomes (CO)

The learners will be able to

1. Understand the basics of programming in C such as tokens, data types, operators etc.
2. Use the Decision making-branching and looping statements in C programming.
3. Handle the concept of arrays and the concept of the user defined functions.
4. Express the uses of structures and pointers
5. Understand and apply the programming concepts of C to problem solving.

CO	**PSO**					
	PSO1	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**
CO1	H	H	H	H	L	H
CO2	H	M	H	M	M	H
CO3	H	M	H	M	M	H
CO4	H	H	H	H	L	H
CO5	H	M	H	H	L	H

CO	**PO**					
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
CO1	H	H	H	M	M	H
CO2	H	H	H	M	M	H
CO3	H	H	H	M	M	H
CO4	H	H	H	M	M	H
CO5	H	H	H	M	M	H

(L-Low, M-Moderate, H-High)

Course Syllabus

Unit I: Overview of C, Constants, Variables and Data types **(12 hours)**

1.1 Basic Structure of C programs – Character set (K1, K2)

1.2 C tokens – Keywords and Identifiers (K1, K2, K3, K4)
1.3 Constants – Variables (K1, K2, K3, K4)
1.4 Data types – Declaration of variables (K1,K2,K3,K4)
1.5 Assigning values to variables (K1, K2, K3, K4)
1.6 Defining symbolic constants – Declaring a variable as constant (K1, K2, K3, K4)
(Chapter 2: Sections 2.8; Chapter 3: Sections 3.2 – 3.8, 3.10 – 3.12)

Unit II: Operators, Expressions, Managing Input and Output Operations (12 hours)

2.1 Introduction-Arithmetic Operators-Relational Operators-Logical Operators (K1, K2, K3, K4)
2.2 Assignment Operators - Increment and Decrement Operators (K1, K2, K3, K4)
2.3 Conditional Operators - Bitwise Operators – Special Operators (K1, K2, K3, K4)
2.4 Arithmetic Expression-Evaluation of Expression-Precedence of Arithmetic Operators (K1, K2, K3, K4)
2.5 Type Conversions in Expressions – Operator Precedence and Associativity (K1, K2, K3, K4)
2.6 Reading a Character-Writing a Character-Formatted Input-Formatted Output (K1, K2, K3, K4)
(Chapter 4: Sections 4.1- 4.12, 4.14, 4.15; Chapter 5: Sections 5.2 – 5.5)

Unit III:Decision Making and Branching, Decision Making and Looping (12 hours)

3.1 Introduction - Decision Making with IF Statement (K1, K2, K3, K4)
3.2 Simple IF – IF ELSE - Nesting of IF ELSE statements (K1, K2, K3, K4)
3.3 The ELSE IF Ladder - The SWITCH statement (K1, K2, K3, K4)
3.4 The conditional (? :) operator- The GOTO statement (K1, K2, K3, K4)
3.5 Introduction - The WHILE statement – The DO statement (K1, K2, K3, K4)
3.6 The FOR statement - Jumps in LOOPS (K1, K2, K3, K4)
(Chapter 6: Sections 6.1 – 6. 9; Chapter 7: Sections 7.1 – 7.5.)

Unit IV: Arrays and User-Defined Functions (12 hours)

4.1 Introduction – One Dimensional Array (K1, K2, K3, K4)
4.2 Declaration and Initialization of One Dimensional Array (K1, K2, K3, K4)
4.3 Two Dimensional Arrays - Initializing Two Dimensional Arrays – Multi Dimensional Arrays (K1, K2, K3, K4)
4.4 Introduction – Need for User-defined functions – A Multi-function Program (K1, K2, K3, K4)
4.5 Elements of user-defined functions – Definition of functions – Return values and their types (K1, K2, K3, K4)
4.6 Function calls – Function declaration - Nesting of functions – Recursion. (K1, K2, K3, K4)
(Chapter 8: Sections 8.1- 8.7; Chapter 10: Sections 10.1 – 10.8, 10. 15, 10.16)

Unit V: Structures and Unions, Pointers (12 hours)

5.1 Introduction-Defining a structure-Declaring structure variables-Accessing structure members (K1, K2, K3, K4)
5.2 Structure initialization-copying and comparing structure variables-Operations on individual members (K1, K2, K3, K4)
5.3 Arrays of structures - Arrays within Structures - StructureswithinStructures - Unions (K1, K2, K3, K4)
5.4 Understanding Pointers – Accessing the address of a variable – Declaring pointer variables (K1, K2, K3, K4)
5.5 Initialization of pointer variables-Accessing a variable through its pointer-Chain of pointers (K1, K2, K3, K4)
5.6 Pointer expressions-Pointer increments and scale factor-Pointers and Arrays (K1, K2, K3, K4)
(Chapter 11: Sections 11. 1 - 11. 10, 11.12; Chapter 12: Sections 12. 2 – 12. 10.)

Text Book:

1. E. Balagurusamy, Programming in ANSI C, 8th Edition, McGraw Hill Education Private Limited, New Delhi, India, 2019.

Reference Book:

1. Ashok N. Kamathne, Programming with C, Pearson Publication, 2009.
2. C: The Complete Reference, Herb Schildt, 4th Edition, Tata McGraw Hill Publishers, 2017

3. Let Us C: Authentic guide to C programming language, YashavantKanetkar, (18th Edition), BPB Publications, 2021

e-Resources:

1. https://nptel.ac.in/
2. www.coursera.org
3. https://swayam.gov.in

SEMESTER – V
UEMAB20 - Elective Practical I: C

Year: III **SEM:** V	**Course Code:** UEMAB20	**Title of the Course:** Elective Practical I : C	**Course Type:** Theory	**Course Category:** Elective	**H/W** 2	**CREDITS** 2	**HOURS** 100

Course Objectives

1. To introduce students to the concept of basic programming, thereby enhancing the logical thinking of the students with regard to programming.
2. To train the students to apply the programming concepts of C to mathematical investigations and problem solving.
3. To construct the ability of students to work independently and do in-depth study of various notions of programming.

Course Outcomes (CO)

The learners will be able to

1. Implement programs with branching and looping statements.
2. Write programs that perform operations using derived data types and functions.
3. Demonstrate a thorough understanding of arrays by designing and implementing programs that search and sort arrays.
4. Perform Matrix operations using C.
5. Use structures and pointers in C programs.

CO	**PSO**					
	PSO1	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**
CO1	H	H	H	H	L	H
CO2	H	M	H	M	M	H
CO3	H	M	H	M	M	H
CO4	H	H	H	H	H	H
CO5	H	M	H	H	L	H

CO	**PO**					
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
CO1	H	H	H	M	M	H
CO2	H	H	H	M	M	H
CO3	H	H	H	M	M	H
CO4	H	H	H	M	M	H
CO5	H	H	H	M	M	H

(L-Low, M-Moderate, H-High)

1. Simple Programs:

a) Sum of ‘n’ natural numbers.
b) Quadratic Equation

c) Simple Interest
d) Mean, Standard deviation and Variance.
e) Generating Prime numbers.
f) Largest of three numbers.

2. Summation of Series:

a) Sin(x)
b) Cos(x)

3. Recursion:

a) nPr and nCr
b) GCD of two numbers.

4. Matrix Manipulation:

a) Addition and Subtraction
b) Transpose.

5. Sorting and Searching:

a) Bubble sort (simple program)
b) Binary search and Median

6. Structures:

Grades of students of a class using structure

Text Book:

1. E. Balagurusamy, Programming in ANSI C, 4th Edition, Tata McGraw – Hill Education Private Ltd. New Delhi, India, 2008.

Reference Books:

1. Ashok N. Kamathne –Programming with C- Pearson publication, 2009.
2. C: The Complete Reference, Herb Schildt, 4th Edition, Tata McGraw Hill Publishers, 2017
3. Let Us C: Authentic guide to C programming language, YashavantKanetkar, (18th Edition), BPB Publications, 2021

e-Resources:

1. https://nptel.ac.in/
2. www.coursera.org
3. https://swayam.gov.in

SEMESTER – VI

UEMAF20 - Object Oriented Programming Using C++

Year: III **SEM:** VI	**Course Code:** UEMAF20	**Title of the Course:** Elective - III: Object Oriented Programming Using C++	**Course Type:** Theory	**Course Category:** Core Elective	**H/W** 4	**CREDITS** 3	**HOURS** 100

Course Objectives

1. To introduce students to the concept of object oriented programming with C++, thereby enhancing the logical thinking of the students with regard to programming.
2. To train the students to apply the programming concepts of C++ to mathematical investigations and problem solving.
3. To construct the ability of students to work independently and do in-depth study of various notions of programming.

Course Outcomes (CO)

The learners will be able to

1. Understand the basics of programming in C++ such as tokens, data types, operators etc.
2. Use the Decision making-branching and looping statements in C++ programming.
3. Handle the concept of arrays and the concept of the user define functions.
4. Express the uses of structures and pointers.
5. Understand and apply the programming concepts of C to problem solving.

CO	**PSO**					
	PSO1	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**
CO1	H	H	H	H	L	H
CO2	H	M	H	M	M	H
CO3	H	M	H	M	M	H
CO4	H	H	H	M	M	H
CO5	H	M	H	M	H	H

CO	**PO**					
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
CO1	H	H	H	M	M	H
CO2	H	H	H	M	M	H
CO3	H	H	H	M	M	H
CO4	H	H	H	M	M	H
CO5	H	H	H	M	M	H

(L-Low, M-Moderate, H-High)

Course Syllabus

Unit I: Principles of OOP and Introduction to C++, Tokens **(12 hours)**

1.1 Basic concepts of object oriented programming – Benefits of OOP (K1, K2)
1.2 Structure of C++ Program - Tokens - Keywords (K1, K2, K3, K4)
1.3 Identifiers and constants - Basic data types (K1, K2, K3, K4)
1.4 User defined data types - Derived data types (K1, K2, K3, K4)

1.5 Symbolic constants – Type compatibility - Declaration of variables (K1, K2, K3, K4)
1.6 Dynamic Initialization of variables – Reference variables (K1, K2, K3, K4)
(Chapter 1: Sections 1.5 -, 1.6; Chapter 2: Sections 2.6; Chapter 3: Sections 3.2 - 3.6, 3.8-3.13)

Unit II: Operators, Expressions and Control Structures (12 hours)

2.1 Operators in C++ - Scope Resolution Operator (K1, K2, K3, K4)
2.2 Member Dereferencing operators – Memory management operators (K1, K2, K3, K4)
2.3 Manipulators – Type cast operator (K1, K2, K3, K4)
2.4 Expressions and their types – Special assignment expressions (K1, K2, K3, K4)
2.5 Implicit conversions – Operator overloading (K1, K2, K3, K4)
2.6 Operator precedence – Control structures (K1, K2, K3, K4)
(Chapter 3: Sections 3.14 - 3. 25)

Unit III: Functions in C++, Classes and Objects (12 hours)

3.1 Introduction- Function prototyping-Call by reference-Return by reference (K1, K2, K3, K4)
3.2 Inline functions-Default arguments-const arguments-Function overloading (K1, K2, K3, K4)
3.3 Specifying a class-Defining member functions-A C++ program with class (K1, K2, K3, K4)
3.4 Making an outside function inline-Nesting of member functions -Private member functions (K1, K2, K3, K4)
3.5 Arrays within a class – Memory allocation for objects – Static data members – Static member functions – Arrays of objects (K1, K2, K3, K4)
3.6 Objects as function arguments – Friendly functions – Returning objects – const member functions – Pointers to members (K1, K2, K3, K4)
(Chapter 4: Sections 4.1, 4.3 – 4. 8, 4.10; Chapter 5: Sections 5.3- 5.18.)

Unit IV: Constructors and Destructors, Operator Overloading (12 hours)

4.1 Introduction – Constructors (K1, K2, K3, K4)
4.2 Parameterized constructors – Multiple constructors in a class (K1, K2, K3, K4)
4.3 Constructors with default arguments – Copy constructor (K1, K2, K3, K4)
4.4 const objects – Destructors (K1, K2, K3, K4)
4.5 Defining operator overloading – Overloading unary operators – Overloading binary operators (K1, K2, K3, K4)
4.6 Overloading binary operators using friends-Rules for overloading operators (K1, K2, K3, K4)
(Chapter 6: Sections 6.1- 6.5, 6.7, 6.10, 6.11; Chapter 7: Sections 7.2 – 7.5, 7.8)

Unit V: Inheritance, Pointers, Managing console I/O Operations (12 hours)

5.1 Introduction – Defining derived classes – Single inheritance (K1, K2, K3, K4)
5.2 Making a private member inheritable – Multilevel inheritance (K1, K2, K3, K4)
5.3 Multiple inheritance – Hierarchical inheritance (K1, K2, K3, K4)
5.4 Hybrid inheritance – Virtual base classes – Abstract classes (K1, K2, K3, K4)
5.5 Pointers – Pointers to Objects – this pointer (K1, K2, K3, K4)
5.6 Introduction – C++ streams – C++ stream classes – Unformatted I/O Operations – Formatted console I/O operations – Managing output with manipulators (K1, K2, K3, K4)
(Chapter 8: Sections 8.1 – 8.10; Chapter 9: Sections 9.2 – 9. 4; Chapter 10: Sections 10.1– 10.6)

Text Book:

1. E. Balagurusamy, Object Oriented Programming with C++, 7th Edition, McGraw Hill Education Private Ltd, New Delhi, India, 2018.

Reference Books:

1. Robert Lafore – Object Oriented Programming in Microsoft C++ - Galgotia Publication, Fourth Edition, 2009.
2. Herbert Schildt – The Complete Reference C++, Tata McGraw Hill Publication, 4th Edition, 2002.
3. Object Oriented Programming in C++, Robert Lafore, 4th Edition, Pearson Publications, 2008.

e-Resources:

1. https://nptel.ac.in/
2. www.coursera.org
3. https://swayam.gov.in

SEMESTER – VI

UEMAG20 - Elective Practical II: C++

Year: III **SEM:** VI	**Course Code:** UEMAG20	**Title of the Course:** Elective Practical II : C++	**Course Type:** Theory	**Course Category:** Elective	**H/W** 2	**CREDITS** 2	**HOURS** 100

Course Objectives

1. To introduce students to the concept of basic programming, thereby enhancing the logical thinking of the students with regard to programming.
2. To train the students to apply the programming concepts of C to mathematical investigations and problem solving.
3. To enhance the ability of students to work independently and do in-depth study of various notions of programming.

Course Outcomes (CO)

The learners will be able to

1. Implement programs with class and constructors.
2. Write programs that perform operations using derived data types and functions.
3. Demonstrate a thorough understanding of arrays by designing and implementing programs that search and sort arrays.
4. Use inheritance properties that promote code reuse in C++.
5. Overload functions and operators in C++.

CO	**PSO**					
	PSO1	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**
CO1	H	H	H	H	L	H
CO2	H	M	H	M	M	H
CO3	H	M	H	M	M	H
CO4	H	H	H	H	L	H
CO5	H	M	H	H	L	H

CO	**PO**					
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**
CO1	H	H	H	M	M	H
CO2	H	H	H	M	M	H
CO3	H	H	H	M	M	H
CO4	H	H	H	M	M	H
CO5	H	H	H	M	M	H

(L-Low, M-Moderate, H-High)

1. Simple program using class and object.
2. Find largest of three numbers using all types of constructors.
3. Calculation of Mean and Standard Deviation.
4. Selection sort.
5. Product of matrices.
6. String manipulation.
7. Operator overloading (Unary)
8. Arrays of Object.
9. Function Overloading.
10. Implementing Inheritance. (Multiple)

Text Book:

1. E. Balagurusamy, Object Oriented Programming with C++, 4th Edition, Tata McGraw – Hill Education Private Ltd. New Delhi, India, 2008.

Reference Books:

1. Robert Lafore – Object Oriented Programming in Microsoft C++ - Galgotia Publication, Fourth Edition, 2009.
2. Herbert Schildt – The Complete Reference C++, Tata McGraw Hill Publication, 4th Edition, 2002.
3. Object Oriented Programming in C++, Robert Lafore, 4th Edition, Pearson Publications, 2008.

e-Resources:

1. https://nptel.ac.in/
2. www.coursera.org
3. https://swayam.gov.in