

**VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY: VISAKHAPATNAM**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**PROGRAM STRUCTURE – VR-20**

**I Year**

**I Semester**

S. No.	Course Code	Course Title	L	T	P	C
1		Mathematics-I	3	0	0	3
2		Applied Chemistry	3	0	0	3
3		Technical English Communication	3	0	0	2
4		Engineering Drawing	1	0	4	3
5		Problem Solving and Programming using C	3	0	0	3
6		Technical English Communication Lab	0	0	3	1.5
7		Applied Chemistry Lab	0	0	3	1.5
8		Problem Solving and Programming using C Lab	0	0	3	1.5
9		Engineering Exploration Lab	0	0	2	1
10		Game, Sports and Yoga	0	0	4	0
<b>Total Credits</b>						<b>19.5</b>

**Legend:** L-Lecture; T-Tutorial; P -Practical; C-Credit

**I Year**

**II Semester**

S. No.	Course Code	Course Title	L	T	P	C
1		Mathematics-II	3	0	0	3
2		Applied Physics	3	0	0	3
3		Object Oriented Programming through C++	3	0	0	3
4		Computer Organization	3	0	0	3
5		Fundamentals of Web Technologies	3	0	0	3
6		Object Oriented Programming through C++ Lab	0	0	3	1.5
7		Applied Physics Lab	0	0	3	1.5
8		Fundamentals of Web Technologies Lab	0	0	3	1.5
9		Constitution of India	2	0	0	0
<b>Total Credits</b>						<b>19.5</b>

**Total Credits (I Year – I&II Sem) = 39**

<b>I Year – II Semester</b>		<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Course Code :</b>	<b>Object Oriented Programming through C++</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES:**

1. To understand how C++ improves C with object-oriented features
2. To learn the syntax and semantics of the C++ programming language.
3. To learn how containment and inheritance promote code reuse in C++.
4. To learn how inheritance and virtual functions implement dynamic binding with polymorphism.
5. To learn how to design and implement generic classes with C++ templates

**COURSE OUTCOMES:**

<b>CO's</b>	<b>At the end of the course, the student will have the ability to:</b>	<b>POs Mapped</b>	<b>Strength of mapping</b>
<b>CO1</b>	Articulate the principles of object-oriented programming and Outline the essential features and elements of the C++ programming language.	PO1	1
<b>CO2</b>	Apply the concepts of class, method, constructor, instance, data abstraction, function abstraction, inheritance, overriding, overloading, and polymorphism.	PO1 PO2	3 3
<b>CO3</b>	Apply virtual and pure virtual function in complex programming situations	PO1 PO2	2 3
<b>CO4</b>	To use template classes and the STL library in C++ and to incorporate exception handling in object oriented concepts	PO2	3

**UNIT- I**

**INTRODUCTION TO C++**

Difference between C and C++- Evolution of C++- The Object Oriented Technology- Disadvantage of Conventional Programming- Key Concepts of Object Oriented Programming- Advantage of OOP- Object Oriented Language. **[6 Hours]**

**UNIT- II**

**CLASSES AND OBJECTS & CONSTRUCTORS AND DESTRUCTOR**

Classes in C++ - Declaring Objects- Access Specifiers and their Scope- Defining Member Function Overloading Member Function- Nested class. Introduction to Constructors and Destructor- Characteristics of Constructor and Destructor-Types of Constructor - Anonymous Objects. **[8 Hours]**

**UNIT- III**

**OPERATOR OVERLOADING AND TYPE CONVERSION & INHERITANCE**

The Keyword Operator- Overloading Unary Operator- Operator Return Type- Rules for Overloading Operators, Overloading Assignment Operator (=). Inheritance, Types of Inheritance. Virtual Base class, object as class member, abstract classes. **[8 Hours]**

**UNIT- IV**

**POINTERS & BINDING POLYMORPHISMS AND VIRTUAL FUNCTIONS**

Pointer, Features of Pointers- Pointer Declaration- Pointer to Class- Pointer Object- this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Introduction- Binding in C++ - Virtual Functions- Rules for Virtual Function- Virtual Destructor. **[8 Hours]**

**UNIT- V**

**GENERIC PROGRAMMING WITH TEMPLATES & EXCEPTION HANDLING**

Generic Programming with Templates, Need for Templates- Definition of class Templates- Normal Function Templates- Overloading of Template Function-Bubble Sort Using Function Templates. Introduction to Standard Template Library: list-set-vector-map-deque. Introduction to Exception Handling: keywords try, throw and catch, multiple catch statements specifying exceptions. **[10 Hours]**

**Text Books:**

1. Programming in C++, Ashok N Kamathane, Pearson 2nd Edition.
2. The Complete Reference C++, Herbert Schildt, TMH.

**Reference Books:**

1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning.
3. Computer Programming with C++,kunal Pimparkhede, cambridge

<b>I Year – II Semester</b>		<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Course Code :</b>	<b>Computer Organization</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES:**

To make students gain knowledge of the basic organization, design, programming of a simple digital computer and to provide an intuition of computer arithmetic, instruction set design, micro programmed control unit, pipelining, vector processing, memory organization and I/O systems.

**COURSE OUTCOMES:**

CO's	At the end of the course, the student will have the ability to:	POs Mapped	Strength of mapping
<b>CO1</b>	Implement computer arithmetic, like addition, subtraction, division and multiplication.	PO1 PO2	3 3
<b>CO2</b>	Design and analyze combinational and sequential circuits.	PO1 PO2	3 3
<b>CO3</b>	Understand the instruction cycle and the Interrupt cycle.	PO1 PO2	2 3
<b>CO4</b>	Understand microprogram and implement it	PO1 PO2 PO4	3 3 3

**UNIT- I**

**Digital Components and Data Representation:** Number Systems, Conversions, BCD, Gray Code, Excess-3 Code, Parity and Hamming Code

**Boolean algebra and minimization:** Boolean expressions and their minimization using algebraic identities; Karnaugh map representation and minimization. **[10 Hours]**

**UNIT- II**

**Combinatorial Circuits:** Design Procedure, Adder, Subtractor, BCD Adder, Multiplexers, De-multiplexers, Encoders and Decoders.

**Sequential Switching Circuits:** Latches, Flip-Flops, Introduction to Register and Counter. **[8 Hours]**

**UNIT- III**

**Computer Arithmetic:** Data representation, Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations.

**Basic Computer Organization and Design:** Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt. **[10 Hours]**

**UNIT- IV**

## Program Structure and Detailed Syllabus (VR 20)

---

**Central Processing Unit:** General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

**Micro programmed Control:** Control memory, Address sequencing, micro program example, design of control unit. **[10 Hours]**

### UNIT- V

**Memory Organization:**Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory. Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access. **[10 Hours]**

#### Text Books:

1. Digital Logic and Computer Design, Moriss Mano, 11th Edition, Pearson Education.
2. Computer System Architecture, 3rd edition, M. Morris Mano, PHI
3. Microprocessor and Interfacing –Douglas V. Hall, 3rd edition, TMH

#### Reference Books:

1. Digital Logic and Computer Design, Moriss Mano, 11th Edition, Pearson Education.
2. Computer System Architecture, 3rd edition, M. Morris Mano, PHI
3. Microprocessor and Interfacing –Douglas V. Hall, 3rd edition, TMH

<b>I Year – II Semester</b>		<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Course Code :</b>	<b>Fundamentals of Web Technologies</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES:**

1. To understand computer programming and application software, package/ suites.
2. Design static web application development and Students will gain the skills and front designs.
3. Able to get project based experience needed for entry into web application and development careers.

**COURSE OUTCOMES:**

<b>CO's</b>	<b>At the end of the course, the student will have the ability to:</b>	<b>POs Mapped</b>	<b>Strength of mapping</b>
<b>CO1</b>	Understand the various applications and computer programming languages purpose.	PO1	1
<b>CO2</b>	Describe the basic concepts of client server application and WWW	PO1 PO2	1 1
<b>CO3</b>	Describe the basic concepts of HTML & CSS to design web pages and web site	PO2 PO3 PO11 PO12	1 2 2 2
<b>CO4</b>	Analyze a given problem and apply requisite appropriate tools for designing interactive web applications	PO2 PO3 PO11 PO12	1 2 2 2

**UNIT- I**

**INTRODUCTION**

**[8 Hours]**

Types of computer applications (Console, Window, web based mobile and cloud applications). Brief History of Internet, What is World Wide Web, Why create a web site, Web Standards. About Client and server process.

**Introduction to HTML:** History of HTML, What are HTML Tags and Attributes? HTML Tag vs. Element, HTML Attributes. Basic Syntax, Standard HTML Document Structure

**UNIT- II**

**HTML Tags:**

**[10 Hours]**

Basic Text Markup, Text formatting tags, Heading types, font tag, Images, image map, Hypertext Links, navigating web pages. What is Lists and various types of list, design the Tables using table tag.

**UNIT- III**

**USER INTERACTIVE WEB PAGE**

**[10 Hours]**

Form tag, user interactive components, Text box, lable, text area, check box, radio button, drop

down box, submit and reset. **Frames:** Importance of frames, divide the web browser window into different sections. Introduction to HTML5.

#### **UNIT- IV**

**Cascading Style Sheets:** **[8 Hours]**

Creating Style Sheet ,CSS Properties, Types of CSS, CSS Styling(Background, Text Format, Controlling Fonts) Working with block elements and objects, Working with Lists and Tables, CSS Id and Class.

#### **UNIT- V**

**Scripting Languages:** **[10 Hours]**

Introduction to Client side and server side scripting languages.

Java Script: Variables, arrays, decision control and loop statements, Functions.

Introduction to PHP script and working with get and post methods.

#### **Text Books:**

1. Programming the World Wide Web, Robert W Sebesta, 7ed, Pearson.
2. Web Technologies, Uttam K Roy, Oxford
3. The Web Warrior Guide to Web Programming, Bai, Ekedahl, Farrelll, Gosselin, Zak, Karparhi, MacIntyre, Morrissey, Cengage

#### **Reference Books:**

1. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
2. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning

<b>I Year – II Semester</b>		<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Course Code :</b>	<b>Object Oriented Programming through C++ Lab</b>	<b>0</b>	<b>0</b>	<b>3</b>	<b>1.5</b>

**COURSE OBJECTIVES:**

1. To strengthen problem solving ability by using the characteristics of an object-oriented approach.
2. To design applications using object oriented features
3. To handle Exceptions in programs.
4. To teach the student to implement object oriented concepts

**COURSE OUTCOMES:**

<b>CO's</b>	<b>At the end of the course, the student will have the ability to:</b>	<b>POs Mapped</b>	<b>Strength of mapping</b>
<b>CO1</b>	Create simple programs using classes and objects in c++ and implement object oriented programs in c++	PO2	3
<b>CO2</b>	Implement object oriented programs using templates and exception handling mechanisms	PO2	3
<b>CO3</b>	Implement programs using STL	PO2 PO3	3 3

**LIST OF EXPERIMENTS**

<b>S.No.</b>	<b>Name of the experiment</b>	<b>Skill</b>
1.	<b>Exercise – 1 (Basics)</b> a) Write a Simple Program on printing “Hello World” and “Hello Name” where name is the input from the user b) Write a C++ program to find both the largest and smallest number in a list of integers c) Write a C++ program to find the sum of individual digits of a positive integer	Input/output
2.	<b>Exercise – 2</b> a) Write a program to implement call by value and call by reference using reference variable. b) Write a program to illustrate scope resolution, new and delete Operators.	Parameter passing techniques, Dynamic Memory Allocation
3.	<b>Exercise – 3</b> a) Write a program illustrating Inline Functions b) Write a program illustrates function overloading. Write 2 overloading functions for power. c) Write a program illustrates the use of default arguments for simple interest function.	Functions



**Program Structure and Detailed Syllabus (VR 20)**

4.	<b>Exercises –4</b> a) Write a program for illustrating Access Specifiers public, private, protected b) Write a program implementing Friend Function c) Write a program to illustrate this pointer	Access Specifiers
5.	<b>Exercise -5</b> a) Write a program to Overload Unary, and Binary Operators as Member Function, and Non Member Function. b) Write a c ++ program to implement the overloading assignment = operator	Operator Overloading
6.	<b>Exercise -6</b> a) Write C++ Programs and incorporating various forms of Inheritance i) Single Inheritance ii) Hierarchical Inheritance iii) Multiple Inheritances iv) Multi-level inheritance v) Hybrid inheritance b) Write a Program in C++ to illustrate the order of execution of constructors and destructors in inheritance	Inheritance
7.	<b>Exercise -7</b> a) Write a program to illustrate runtime polymorphism b) Write a program illustrates pure virtual function and calculate the area of different shapes by using abstract class.	Polymorphism
8.	<b>Exercise -8</b> a) Write a C++ Program to illustrate template class b) Write a Program to illustrate class templates with multiple parameters c) Write a Program to illustrate member function templates	Templates
9.	<b>Exercise -9</b> a) Write a Program to implement List and List Operations. b) Write a Program to implement Vector and Vector Operations. c) Write a Program to implement Deque and Deque Operations. d) Write a Program to implement Map and Map Operations. e) Write a Program to implement set and set Operations.	STL
10.	<b>Exercise -10</b> a) Write a Program Containing a Possible Exception. Use a Try Block to Throw it and a Catch Block to Handle it Properly. b) Write a Program to Demonstrate the Catching of All Exceptions.	Exception Handling

**Text Books:**

1. The Complete Reference C++, Herbert Schildt, TMH.
2. Programming in C++, Ashok N Kamathane, Pearson 2nd Edition.

**Reference Books:**

1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning.
3. Computer Programming with C++,kunalPimparkhede,cambridge

<b>I Year – II Semester</b>		<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>Course Code :</b>	<b>Fundamentals of Web Technologies Lab</b>	<b>0</b>	<b>0</b>	<b>3</b>	<b>1.5</b>

**COURSE OBJECTIVES:**

1. To understand computer programming and application software, package/ suites.
2. Design static web application development and Students will gain the skills and front designs.
3. Able to get project based experience needed for entry into web application and development careers.

**COURSE OUTCOMES:**

<b>CO's</b>	<b>At the end of the course, the student will have the ability to:</b>	<b>POs Mapped</b>	<b>Strength of mapping</b>
<b>CO1</b>	Understand the usage and designing of web pages using HTML & CSS	PO1	1
<b>CO2</b>	Able to design the user interactive pages and web page layouts	PO2 PO3	2 2
<b>CO3</b>	Analyze a given problem and apply requisite appropriate tools for designing interactive web applications	PO5 PO11 PO12	2 3 2

**LIST OF EXPERIMENTS**

<b>S.No.</b>	<b>Name of the experiment</b>	<b>Skill</b>
1.	<b>Exercise – 1</b> Design Web page to illustrate the following <ul style="list-style-type: none"> <li>• Title of web page</li> <li>• Heading styles</li> <li>• Various Text formatting tags</li> </ul>	Web Page Design
2.	<b>Exercise – 2</b> Design Web page to illustrate the following <ul style="list-style-type: none"> <li>• Apply font tag to the text</li> <li>• Upload and resize the image</li> <li>• Implement Image maps</li> </ul>	Web Page Design
3.	<b>Exercise – 3</b> Design Web page to illustrate the following <ul style="list-style-type: none"> <li>• Various List types</li> <li>• Display the class time table using table tag</li> </ul>	Web Page Design
4.	<b>Exercise – 4</b> Design Web page to illustrate the following <ul style="list-style-type: none"> <li>• Web page navigation (self and new page)</li> </ul>	Web Page Design

## Program Structure and Detailed Syllabus (VR 20)

	<ul style="list-style-type: none"><li>• Implement image as web page navigation</li></ul>	
5.	<b>Exercise – 5</b> Implement the various CSS <ul style="list-style-type: none"><li>• Inline CSS</li><li>• Internal CSS</li><li>• External CSS</li></ul>	CSS
6.	<b>Exercise – 6</b> Design the Login and Registration forms and apply CSS	Web Page Design
7.	<b>Exercise – 7</b> Java script to implement decision control and loop statements	Web Page Design
8.	<b>Exercise – 8</b> Java script to implement functions concepts	Scripting
9.	<b>Exercise – 9</b> Login form validation using java script	Validation
10.	<b>Exercise – 10</b> Working with get and post method mechanism to interact server using PHP script	Web Page Design

### Text Books:

1. Programming the World Wide Web, Robert W Sebesta, 7ed, Pearson.
2. Web Technologies, Uttam K Roy, Oxford
3. The Web Warrior Guide to Web Programming, Bai, Ekedahl, Farrell, Gosselin, Zak, Karparhi, MacIntyre, Morrissey, Cengage

### Reference Books:

1. Web Technologies, HTML < JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
2. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning