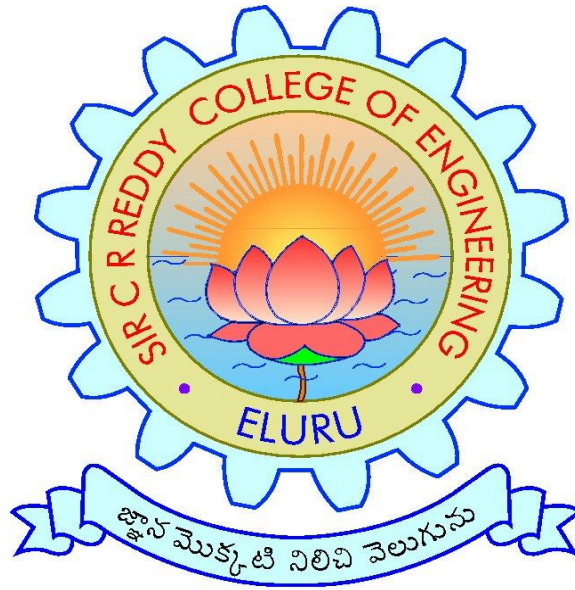


**SIR C R REDDY COLLEGE OF ENGINEERING, ELURU**  
**DEPARTMENT OF INFORMATION TECHNOLOGY**

# **OBJECT ORIENTED PROGRAMMING COURSE**

## **HANDOUT**



**SUBJECT: OBJECT ORIENTED PROGRAMMING**

**CLASS: II/IV B.Tech (A & B sections )Semester-I, A.Y.2023-2024**

**INSTRUCTOR: Dr K Satyanarayana**

### Course Handout Index

S. No	Description
1	College Vision & Mission
2	Department Vision & Mission
3	Program Educational Objectives (PEOs)
4	Program Outcomes (POs)
5	Program Specific Outcomes (PSOs)
6	JNTUK Academic Calendar
7	Department Academic Calendar
8	Course Description
9	Course Objectives
10	Course Outcomes
11	Syllabus
12	Lesson Plan
13	Evaluation Pattern
14	Timetable
15	Unit wise Questions and Programs

### **College Vision & Mission**

**Vision:** To emerge as a premier institution in the field of technical education and research in the state and as a home for holistic development of the students and contribute to the advancement of society and the region.

**Mission:** To provide high quality technical education through a creative balance of academic and industry oriented learning; to create an inspiring environment of scholarship and research; to instill high levels of academic and professional discipline; and to establish standards that inculcate ethical and moral values that contribute to growth in career and development of society in general.

### **Department Vision & Mission**

**Vision:** To be a premier department in the region in the field of Information Technology through academic excellence and research that enable graduates to meet the challenges of industry and society.

**Mission:** To Provide dynamic teaching-learning environment to make the students industry ready and advancement in career; to inculcate professional and leadership quality for better employability and entrepreneurship; to make high quality professional with moral and ethical values suitable for industry and society.

### **Program Educational Objectives (PEOs)**

**PEO1:** Solve real world problems through effective professional skills in Information Technology industry and academic research.

**PEO2:** Analyze and develop applications in Information Technology domain and adapt to changing technology trends with continuous learning.

**PEO3:** Practice the profession in society with ethical and moral values.

## **Program Outcomes (POs)**

**PO1: Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

**PO2: Problem Analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using the first principles of mathematics, natural sciences, and engineering sciences.

**PO3: Design/Development of Solutions:** Design solutions for complex engineering problems and system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, society, and environmental considerations.

**PO4: Conduct Investigations of Complex Problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

**PO5: Modern Tool Usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

**PO6: The Engineer and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

**PO7: Environment and Sustainability:** Understand the impact of the professional engineering solutions in society and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

**PO8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

**PO9: Individual and Team Work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multi-disciplinary settings.

**PO10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

**PO11: Project Management and Finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multi-disciplinary environments.

**PO12: Life-long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**Program Specific Outcomes (PSOs)**

**PSO1: Design Skill:** Design and develop softwares in the area of relevance under realistic constraints.

**PSO2: New Technology:** Adapt new and fast emerging technologies in the field of Information Technology.

## JNTUK Academic Calendar

Website: www.jntuk.edu.in  
Email: dap@jntuk.edu.in



Phone: 0884-2300991

**Directorate of Academic Planning**  
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA  
KAKINADA-533003, Andhra Pradesh, INDIA  
(Established by AP Government Act No. 30 of 2008)

Lt. No. DAP/AC/II Year - B, Tech/2023

Date 05.08.2023

**Dr. KVSG Murali Krishna,**  
*M.E., Ph.D.,*  
**Director, Academics & Planning**  
**JNTUK, Kakinada**

To  
All the Principals of Affiliated Colleges,  
JNTUK, Kakinada.

### Academic Calendar for II Year - B. Tech for the AY 2023-24

I SEMESTER			
Description	From	To	Weeks
Commencement of Class Work	07.08.2023		
I Unit of Instruction	07.08.2023	30.09.2023	8W
I Mid Examinations	23.09.2023	30.09.2023	
II Unit of Instructions	02.10.2023	25.11.2023	8W
II Mid Examinations	20.11.2023	25.11.2023	
Preparation & Practicals	27.11.2023	09.12.2023	2W
End Examinations	11.12.2023	23.12.2023	2W
Commencement of II Semester Class Work	27.12.2023		
II SEMESTER			
I Unit of Instructions	27.12.2023	17.02.2024	8W
I Mid Examinations	12.02.2024	17.02.2024	
II Unit of Instructions	19.02.2024	13.04.2024	8W
II Mid Examinations	08.04.2024	13.04.2024	
Preparation & Practicals	15.04.2024	27.04.2024	2W
End Examinations	29.04.2024	11.05.2024	2W
Summer Internship	13.05.2024	06.07.2024	8W
Commencement of III- I Class Work	08.07.2024		

*KVSG* Director  
Academic Planning  
JNTUK Kakinada  
Academics & Planning,  
JNTUK

Copy to the Secretary to the Hon'ble Vice Chancellor, JNTUK  
Copy to Rector, JNTUK  
Copy to Registrar, JNTUK  
Copy to Director Academic Audit, JNTUK  
Copy to Director of Evaluation, JNTUK



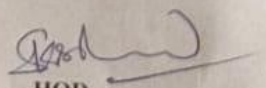
## SIR C R REDDY COLLEGE OF ENGINEERING

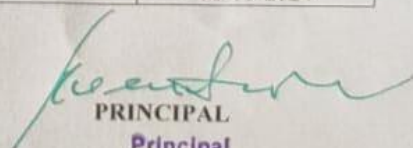
ELURU-534007, WEST GODAVARI DIST, ANDHRA PRADESH, INDIA  
(Approved by AICTE, New Delhi & Permanently affiliated to JNTUK, Kakinada)  
Telephone No: 08812-230840, 230565, Fax: 08812-224193  
Website: www.sircrrengg.ac.in

# IQAC

### DEPARTMENT OF INFORMATION TECHNOLOGY II/IV ACADEMIC CALENDAR 2023 – 2024

EVENTS / ACTIVITIES	I- SEMESTER	II- SEMESTER
Registration of Credits/Electives	15-07-2023 to 5-07-2023	10-12-2023 To 24-12-2023
Commencement of classes	7-08-2023	27-12-2023
Class work – 1 <sup>st</sup> Phase of Instruction (From... To...)	07-08-2023 To 30-09-2023	27-12-2023 To 17-02-2024
Class Review Committee Meeting-I/Parent-Teachers Meet	September 2023	February 2024
Guest Lecture/Seminar/Workshop	September 2023	February 2024
Assignment - I	10-09-2023	01-01-2023
MID Examination – I & Quiz - I	25-09-2023 To 30-09-2023	12-02-2024 To 17-02-2024
Mid-Semester Feedback	1-10-2023	18-02-2024
Last date for display of Marks/Answer Scripts	8-10-2023	25-02-2024
Class work – 2 <sup>nd</sup> Phase of Instruction (From... To...)	02-10-2023 To 25-11-2023	19-02-2024 To 13-04-2024
Remedial classes	After 1 <sup>st</sup> MID	After 1 <sup>st</sup> MID
Class Review Committee Meeting-II	November 2023	April 2024
Guest Lecture/Seminar/Workshop	November 2023	March 2024
Assignment - II	01-11-2023	22-03-2024
MID Examination – II & Quiz - II	20-11-2023 To 25-11-2023	08-04-2024 To 13-04-2024
Class work last working day	18-11-2023	05-04-2024
End-Semester Feedback & Course End Survey	26-11-2023	14-04-2024
Last date for display of Marks/Answer Scripts	30-11-2023	21-04-2024
Preparation holidays and Semester End Practical Examinations	27-11-2023 To 09-12-2023	15-04-2024 To 27-04-2024
Semester End Theory Examinations	11-12-2023 To 23-12-2023	29-04-2024 To 11-05-2024
Summer Internship	--	13-05-2024 To 06-07-2024

  
**HOD**  
HEAD OF THE DEPARTMENT  
Information Technology  
Sir C.R.R. College of Engg.  
ELURU-534 007

  
**PRINCIPAL**  
Principal  
Sir C.R.R. College of Engineering  
ELURU - 534 007

### **Course description:**

OOP through C++ is a programming language with a simple syntax and a powerful set of libraries. Object-oriented programming – As the name suggests uses objects in programming. Object-oriented programming aims to implement real-world entities like inheritance, hiding, polymorphism, etc in programming. It is an interpreted language, with a rich programming environment, including a robust debugger and profiler. While it is easy for beginners to learn.

### **Scope and objectives:**

- Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects
- Understand dynamic memory management techniques using pointers, constructors, destructors
- Describe the concept of function overloading, operator overloading, virtual functions and polymorphism
- Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming
- Demonstrate the use of various OOPs concepts with the help of programs

### **Prerequisite:**

C++ is an object-oriented programming language while C is just a basic programming language. So, you can learn the programming concepts in the C language in a relatively easier way. C++ is a more composite language and is the superset of C. Thus, a perceptive of the C language facilitates a rapid and easy learning of C++.

### **Course Outcomes**

After the completion of the course, student will be able to

<b>CO</b>	<b>CO Description</b>	<b>Level</b>
CO1	Understand Object oriented concepts to solve the problems	L2
CO2	Build C++ Classes using appropriate encapsulation and design principles	L3
CO3	Apply inheritance and polymorphism using C++	L3
CO4	Apply Exception Handling and Templates using C++	L3



## Syllabus

### 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.

### 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, Constructors and Destructors, Introduction, Constructors and Destructor- Characteristics of Constructor and Destructor, Application with Constructor, Constructor with Arguments (parameterized Constructor, Destructors- Anonymous Objects.

### UNIT III

Operator Overloading and Type Conversion & Inheritance: The Keyword Operator, Overloading Unary Operator, Operator Return Type, Overloading Assignment Operator (=), Rules for Overloading Operators, Inheritance, Reusability, Types of Inheritance, Virtual Base Classes, Object as a Class Member, Abstract Classes, Advantages of Inheritance-Disadvantages of Inheritance.

### UNIT IV

Pointers & Binding Polymorphisms and Virtual Functions: Pointer, Features of Pointers, Pointer Declaration, Pointer to Class, Pointer Object, The this Pointer, Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Binding in C++, Virtual Functions, Rules for Virtual Function, Virtual Destructor.

### UNIT V

Generic Programming with Templates, Need for Templates, Definition of class Templates, Normal Function Templates, Overloading of Template Function, Bubble Sort Using Function Templates, Difference Between Templates and Macros, Linked Lists with Templates, Exception Handling, Principles of Exception Handling, The Keywords try throw and catch, Multiple Catch Statements – Specifying Exceptions.

#### Text Books:

- 1) A First Book of C++, Gary Bronson, Cengage Learning.
- 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) Programming in C++, Ashok N Kamthane, Pearson 2<sup>nd</sup> Edition.

#### e- Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105151/>
- 2) <https://github.com/topics/object-oriented-programming>

## Lesson Plan

Unit	Topics	Teaching Aids	CO
I	Introduction- JAVA	BB/PPT	1
	Introduction to C++: Difference between C and C++, Evolution of C++	BB/PPT	1
	The Object-Oriented Technology, Disadvantage of Conventional Programming	BB/PPT	1
	Key Concepts of Object-Oriented Programming	BB/PPT	1
	Advantage of OOP, Object Oriented Language.	BB/PPT	1
II	Classes in C++-Declaring Objects	BB/PPT	2
	Access Specifiers and their Scope, Defining Member Function-	BB/PPT	2
	Overloading Member Function, Nested class,	BB/PPT	2
	Constructors and Destructors	BB/PPT	2
	Constructors and Destructor- Characteristics of Constructor and Destructor	BB/PPT	2
	Application with Constructor, Constructor with Arguments (parameterized Constructor,	BB/PPT	2
	Destructors- Anonymous Objects.	BB/PPT	2
III	The Keyword Operator, Overloading Unary Operator	BB/PPT	1
	Operator Return Type, Overloading Assignment Operator (=)	BB/PPT	1
	Rules for Overloading Operators and examples	BB/PPT	3
	Inheritance, Reusability, real time examples	BB/PPT	3
	Types of Inheritance and Examples	BB/PPT	3
	Virtual Base Classes	BB/PPT	3
	Object as a Class Member	BB/PPT	3
	Abstract Classes	BB/PPT	3

	Advantages of Inheritance-Disadvantages of Inheritance.	BB/PPT	3
IV	Pointer, Features of Pointers	BB/PPT	3
	Pointer Declaration, Pointer to Class	BB/PPT	3
	Pointer Object, The this Pointer	BB/PPT	3
	Pointer to Derived Classes and Base Class and examples	BB/PPT	3
	Binding Polymorphisms and Virtual Functions and examples	BB/PPT	3
	Binding in C++, Virtual Functions and examples	BB/PPT	3
	Rules for Virtual Function, Virtual Destructor and examples	BB/PPT	3
V	Generic Programming with Templates,	BB/PPT	4
	Need for Templates, Definition of class Templates	BB/PPT	4
	Normal Function Templates, Overloading of Template Function and examples	BB/PPT	4
	Bubble Sort Using Function Templates	BB/PPT	4
	Difference Between Templates and Macros	BB/PPT	4
	Linked Lists with Templates	BB/PPT	4
	Exception Handling, Principles of Exception Handling and examples	BB/PPT	4
	The Keywords try throw and catch and examples	BB/PPT	4
Multiple Catch Statements – Specifying Exceptions.	BB/PPT	4	

**Evaluation Pattern**

S. No	Components	Internal	External	Total
1	Theory	30	70	100
2	Engineering Graphics/Design/Drawing	30	70	100
3	Practical	15	35	50
4	Mini Project/Internship/Industrial Training/ Skill Development programmes/Research Project	-	50	50
5	Project Work	60	140	200

Marks Range Theory (Max – 100)	Marks Range Lab (Max – 50)	Level	Letter Grade	Grade Point
≥ 90	≥ 45	Outstanding	A+	10
≥80 to <89	≥40 to <44	Excellent	A	9
≥70 to <79	≥35 to <39	Very Good	B	8
≥60 to <69	≥30 to <34	Good	C	7
≥50 to <59	≥25 to <29	Fair	D	6
≥40 to <49	≥20 to <24	Satisfactory	E	5
<40	<20	Fail	F	0
-		Absent	AB	0

### Timetable

Day/Time	09.00-09.50	09.50-10.40	11.00-11.50	11.50-12.40	01.40-02.30	02.30-03.20	03.20-04.10	04.10-05.00
<b>Mon</b>		A		B	OOPS LAB A SECTION			
<b>Tue</b>		A					B	
<b>Wed</b>		B		A	OOPS LAB B SECTION			
<b>Thu</b>		B						
<b>Fri</b>	B		A					
<b>Sat</b>	A				*****			

# UNIT WISE QUESTIONS

## **Unit-1**

### **Introduction to C++**

1. Describe the following characteristics of OOP
  - i Encapsulation
  - ii Polymorphism,
  - iii Inheritance
2. Define the 'this' pointer, with an example, indicate the steps involved in referring to members of the invoking object.
3. Discuss the issues of procedure oriented systems with respect to object oriented systems?
4. Give the comparison of C and C++ with examples..
5. What are pointers explain with an example. . .
6. What is function overloading give example?.
7. Differentiate between procedure oriented and object oriented programming.
8. Explain inline functions?

## **Unit-2**

### **Classes & constructors**

1. Write a C++ program to count the number of objects of a certain class.
2. What is a class? How is it created? Write an example class.
3. What are constructors? How are they different from member functions?
4. What are static data members? Explain with an example what the use of static data members is.
5. Demonstrate with C++ program for
  - i) Passing objects to functions
  - ii) Returning objects
6. Explain the features of new and delete?

7. What is the benefit of copy constructor? Explain the necessity of defining your own copy constructor?.
8. What is a friend function? Why is it required? Explain with an example.
9. What is the use of operator overloading? Write a program to overload post and pre increment operators.
10. Explain Generic function with example.

## **Unit-3**

### **Inheritance**

1. Explain different types of inheritance with block diagram and an example for each
2. What is the ambiguity that arises in multiple inheritance? How it can be overcome. Explain with example.
3. Discuss with examples, the implications of deriving a class from an existing class by the 'public' and 'protected' access specifiers.
4. Write a c++ program to initialize base class members through a derived class constructor..
5. What is inheritance? How to inherit a base class as protected? Explain it in Multiple base classes?
6. With an example explain, multiple base class inheritance?
7. is the need of virtual function? With an example, explain overriding of Member function of base in derived class?
8. What is the virtual destructor?
9. List the library classes that handle streams in c++.
10. When Constructors and Destructors Are Executed.
11. Explain Granting Access.
12. What are virtual functions? What is the use. Give an example. How compilers resolve a function call.
13. Describe briefly with a figure, class hierarchy provided by c++ for stream handling.

14. Define and give the syntax for the following.

- a. Virtual function
- b. Pure Virtual function
- c. Abstract Base Class

## **Unit-4**

### **Virtual functions and Polymorphism**

1. Why friend function is required to overload binary operators?
2. What are the rules for overloading operators?
3. Write the difference between Early and Late Binding.
4. Explain Pure Virtual Functions.
5. Explain Calling a Virtual Function Through a Base Class Reference.
6. What are the rules for overloading the operator?
7. Define a class Date, use overloaded + operator to add two dates and display the result. Assume non leap year dates.

## **Unit-5**

### **Exception Handling, STL**

1. What are the new style cast operators explain the syntax of these operators with example ?
2. What are class templates.? How are they created? What is the need for class templates? Create a template for bubble sort functions.
3. Explain the C++ style solution for handling exceptions
4. Explain try catch and throw exception handling in c++
5. Explain different types of type conversion.
6. Explain with example, how Function Templates are implemented?
7. With an example, explain how to overload pointer to member operator
8. Define a function template giving its syntax. Write a c++ program to implement array representation of a stack for integers, characters and

floating point numbers using class template.

9. Explain new and delete operators overloading in c++ with examples?