MIT | Academy of Engineering (An Autonomous Institute Affiliated to Savitribai Phule Pune University)	COURSE SYLLABUS	
SCHOOL OF COMPUTER ENGINEERING	W.E.F.	AY: 2023 - 2024 (Rev. 2022)
SECOND YEAR BACHELOR OF TECHNOLOGY	COURSE NAME	Problem Solving Using Object Oriented Programming (C++)
	COURSE CODE	2304261
	COURSE CREDITS	2
RELEASE DATE : 01/07/2023	REVISION NO.	0.0

TEACHING SCHEME (HOURS/WEEK)		EXAMINATION SCHEME AND MARKS					
		THEORY			LABORATORY		TOTAL
LECTURE	PRACTICAL	IA	MSE	ESE	CA	PRACT/DEMO/PRES	
NIL	4	NIL	NIL	NIL	35	40	75

PREREQUISITE : BASIC COMPUTING KNOWLEDGE

COURSE OBJECTIVES :

2304261.CEO.1: To demonstrate Object Oriented Programming concepts.
2304261.CEO.2: To illustrate modularity, scalability and code reusability.
2304261.CEO.3: To identify various perspectives of a problem.
2304261.CEO.4: To choose the optimal solution for challenging problems.

COURSE OUTCOMES :

After successful completion of the course, students will be able to,

2304261.CO.1: Develop solutions for real world problems using Object Oriented Programming. [L3]
2304261.CO.2: Choose suitable programming concepts to reduce complexity, and enhance productivity. [L3]
2304261.CO.3: Apply critical thinking and programming skills for problem solving. [L3]
2304261.CO.4: Utilize logic building traits efficiently for solving challenging problems. [L3]

COURSE ABSTRACT:

Problem Solving using Object Oriented Programming (C++) is the Skill Development Course. The basic aim of the course is to introduce the students to Object Oriented Programming using C++ and make their concepts clear for designing real time applications using the same. The inclusion of Challenging Problems in the course will enhance students' logical thinking and boost their analytical skills. Various stages in problem solving will help students to deal with complex problems easily. This course will improve students problem solving abilities and coding skills.

CONTENTS :

Introduction to C++, Tokens, Data types, Operators, Structure of C++ Programs, Classes and Objects, Constructors, Destructors; Arrays, Expressions and Control Structures, Types of Constructors and Member Functions; Overloading; Reference and Pointers, Dynamic Memory Management; Functions in C++: Argument passing in function, Inline Functions, Default Arguments, Constant Arguments, Friend class, Friend function; Inheritance; Polymorphism; Exception Handling; std::move in utility; Templates; Standard Template Library; Type casting and cast operators.

Self Study: Structures, Union

Further Readings: File Handling Operations, Multithreading

OBJECT ORIENTED PROGRAMMING:

PRACTICAL: All assignments should be performed considering real world applications.

PRACTICAL NO.01	**CLASS AND OBJECTS**	**2 HOURS**
Write a program on class, constructor, destructor, and user-defined functions.		
PRACTICAL NO.02	**CONSTRUCTOR TYPES**	**2 HOURS**
Implement various types of constructors and member functions.		
PRACTICAL NO.03	**OPERATOR OVERLOADING**	**2 HOURS**
Write a program on operator overloading using default and parameterized constructors.		
PRACTICAL NO.04	**DYNAMIC MEMORY**	**4 HOURS**
a) Assignment based on references and pointers. b) Write a program on new and delete operators in C++ for dynamic memory.		
PRACTICAL NO.05	**FUNCTION TYPES, FRIEND CLASS**	**4 HOURS**
Assignment based on class, objects, this pointer, inline function, static member function and friend class.		

PRACTICAL NO.06	**INHERITANCE**	**2 HOURS**
Implement different types of inheritance.		
PRACTICAL NO.07	**POLYMORPHISM**	**2 HOURS**
Write a program on polymorphism.		
PRACTICAL NO.08	**EXCEPTION HANDLING**	**2 HOURS**
Implement exception handling operations.		
PRACTICAL NO.09	**STD::MOVE IN UTILITY**	**2 HOURS**
Write a program on std::move in utility.		
PRACTICAL NO.10	**TEMPLATES**	**2 HOURS**
Implement class template, function template.		
PRACTICAL NO.11	**STANDARD TEMPLATE LIBRARY**	**4 HOURS**
a) Assignment based on list class as a container of standard template library. b) Assignment based on vector class as a container of standard template library.		
PRACTICAL NO.12	**TYPE CASTING**	**2 HOURS**
Assignment based on type casting and casting operators.		

PROBLEM SOLVING:		
01	**BASIC PROGRAMMING**	**2 HOURS**
02	**NUMBER SYSTEM**	**2 HOURS**
03	**ARRAY**	**2 HOURS**
04	**MATRIX**	**2 HOURS**
05	**STRING**	**2 HOURS**
06	**MATH**	**2 HOURS**
07	**BIT MANIPULATION**	**2 HOURS**
08	**SEARCHING AND SORTING**	**2 HOURS**
09	**INHERITANCE**	**2 HOURS**
10	**POLYMORPHISM**	**2 HOURS**
11	**TEMPLATES/COLLECTION**	**2 HOURS**
12	**MULTITHREADING**	**4 HOURS**
13	**STACK AND QUEUE (FURTHER PRACTICE)**	**2 HOURS**
14	**FILE PROCESSING (FURTHER PRACTICE)**	**2 HOURS**

EXAMINATION SCHEME

1. Continuous Assessment: 20 Marks
2. Mid Semester Practical Examination for Problem Solving: 15 Marks
3. End Semester Practical Examination for OOP(C++): 20 Marks
4. End Semester Practical Examination for Problem Solving: 20 Marks

TEXT BOOKS

1. E. Balgurusawmy. (2020). *Object Oriented Programming with C++.* 8^{th} Edition, McGraw Hill Publishers. ISBN 9389949181.
2. Paul Deitel, Harvey Deitel. (2017). *C++: How to Program.* 10^{th} Edition, Pearson International. ISBN 9789332585737.
3. Sprankle Maureen. (2011). *Problem Solving and Programming Concepts.* 9^{th} Edition, Person. ISBN 978-0132492645.
4. Savitch Walter. (2017). *Problem Solving with C++.* 10^{th} Edition, Person. ISBN 978-0134448282.

REFERENCE BOOKS

1. Bjarne Stroustroup. (2014). *Programming - Principles and Practice using C++*. 2^{nd} Edition, Addison-Wesley Educational Publishers Inc . ISBN 978-0275967819.
2. Herbtz Schildt. (2017). *C++: The Complete reference*. 4^{th} Edition, McGraw Hill Education. ISBN 978-0070532465.
3. V. Anton Spraul. (2012). *Think Like a Programmer: An Introduction to Creative Problem Solving*. 1^{st} Edition, No Starch Press US. ISBN 978-1593274245.