GUJARAT TECHNOLOGICAL UNIVERSITY (GTU)

Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021)
Semester - II

Course Title: Basic Object Oriented Programming
(Course Code: 4320702)

Diploma programme in which this course is offered	Semester in which offered
Computer Engineering	Second

1. RATIONALE

This course intends to teach the students about basic concepts of Object-Oriented Programming (OOP) and C++. Large programs are probably the most complicated entities ever created by humans. Because of this complexity, programs are prone to error and software errors can be expensive and even life-threatening. Object-oriented programming offers a new and powerful way to cope with this complexity and act as the backbone to all other courses that are based on Object Oriented concept. Therefore by learning this course senicierly the students will be able to develop programs in ‘C++’ using Object Oriented Programming Concepts.

2. COMPETENCY

The course content should be taught and implemented with the aim to develop various types of related skills leading to the achievement of the following competency

- **Develop programs in ‘C++’ using Object Oriented Programming Concepts.**

3. COURSE OUTCOMES (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with this competency are to be developed in the student to display the following COs:

The practical experiences and relevant soft skills associated with this course are to be taught and implemented, so that the student demonstrates the following industry-oriented COs associated with the above-mentioned competency:

a) Select procedural oriented and object-oriented approach to solve given problem.
b) Implement object-oriented program using constructor and destructor.
c) Implement Inheritance for code reuse in c++ program.
d) Develop program using runtime polymorphism.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (CI+T/2+P/2)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
CI	T	P	C	CA	ESE	CA	ESE	
3	0	4	5	70	30	25	25	150

Out of 30 marks under the theory CA, 10 marks are for assessment of the micro-project to facilitate integration of COs and the remaining 20 marks is the average of 2 tests to be taken during the semester for the assessing the attainment of the cognitive domain UOs required for the attainment of the COs.

Legends: CI-ClassRoom Instructions; T – Tutorial/Teacher Guided Theory Practice; P - Practical; C – Credit, CA - Continuous Assessment; ESE - End Semester Examination.

5. SUGGESTED PRACTICAL EXERCISES

The following practical outcomes (PrOs) that are the sub-components of the COs. *Some of the* ***PrOs*** *marked* ***'*'*** *are compulsory, as they are crucial for that particular CO. These PrOs need to be attained at least at the 'Precision Level' of Dave's Taxonomy related to 'Psychomotor Domain'.*

S.No.	**Practical Outcomes (PrOs)**	**Unit No.**	**Approx. Hrs. required**
1	Create your account on github and save simple C++ program in github.	I	02
2	Develop minimum 5 programs using cin and cout.	I	02
3	Develop programs using scope resolution operator, simple manipulators, and enumeration.	I	02
4	Develop programs using call by reference and return by reference, default arguments, constant arguments, inline and function overloading.	II	06
5	Develop programs using structures.	II	02
6	Define minimum 5 different classes such as student, distance, shape, employee, account, inventory, vector, movie-ticket booking, time, point, etc. with data member & member functions. Also Develop programs to test those classes functionality.	II	10
7	Develop Programs using array of objects and static member function	II	04
8	Develop programs to pass object as an argument and Returning object.	II	04
9	Develop programs using friend function and Friend class.	II	04
10	Apply the concepts of constructors and destructors in the Programs developed in unit-2 and test those programs.	III	06
11	Develop programs using single, multilevel, multiple Inheritance.	IV	06
12	Develop programs using Constructors in base and derive classes.	IV	02
13	Develop a program to show use of this pointer.	V	02
14	Develop a program using runtime polymorphism.	V	02
15	Develop at least 2 programs using file operations.	V	02
	Total		56

Note

i. *More* ***Practical Exercises*** *can be designed and offered by the respective course teacher to develop the industry relevant skills/outcomes to match the COs. The above table is only a suggestive list.*

ii. *The following are some* ***sample*** *'Process' and 'Product' related skills (more may be added/deleted depending on the course) that occur in the above listed* ***Practical Exercises*** *of this course required which are embedded in the COs and ultimately the competency..*

S. No.	Sample Performance Indicators for the PrOs	Weightage in %
1	Correctness of program	30
2	Readability and documentation of the program/Quality of input and output displayed (messaging and formatting)	10
3	Code efficiency	20
4	Debugging ability	20
5	Program execution/answer to sample questions	20
	Total	**100**

6. MAJOR EQUIPMENT/ INSTRUMENTS AND SOFTWARE REQUIRED

These major equipment/instruments and Software required to develop PrOs are given below with broad specifications to facilitate procurement of them by the administrators/management of the institutes. This will ensure conduction of practical in all institutions across the state in proper way so that the desired skills are developed in students.

S. No.	Equipment Name with Broad Specifications	PrO. No.
1	Computer with latest configuration with windows or unix os	All
2	C++ Compiler	All

7. AFFECTIVE DOMAIN OUTCOMES

The following ***sample*** Affective Domain Outcomes (ADOs) are embedded in many of the above-mentioned COs and PrOs. More could be added to fulfil the development of this competency.

a) Follow safety practices.
b) Practice good housekeeping.
c) Demonstrate working as a leader/a team member.
d) Maintain tools and equipment
e) Follow ethical practices.

The ADOs are best developed through the laboratory/field-based exercises. Moreover, the level of achievement of the ADOs according to Krathwohl's 'Affective Domain Taxonomy' should gradually increase as planned below:

i. 'Valuing Level' in 1st year
ii. 'Organization Level' in 2nd year.
iii. 'Characterization Level' in 3rd year.

8. UNDERPINNING THEORY

The major Underpinning Theory is formulated as given below and only higher level UOs of *Revised Bloom's taxonomy* are mentioned for development of the COs and competency in the students by the teachers. (Higher level UOs automatically includes lower level UOs in them). If required, more such higher level UOs could be included by the course teacher to focus on attainment of COs and competency.

Unit	Unit Outcomes (UOs)	Topics and Sub-topics
Unit –1: Principles of Object Oriented Programming	1a. Differentiate procedure and object oriented languages 1b. Explain the general structure of C++ 1c. Develop program using cin and cout 1d. Develop program using scope resolution operator, manipulator and enumeration	1.1 Overview of Structure Programming language 1.2 The object-oriented Approach 1.3 Basic Concept of object-oriented Programming:-object, class, Inheritance, Data abstraction, encapsulation, polymorphism, dynamic binding, message passing 1.4 Advantages of Object Oriented Programming 1.5 Usage of Object Oriented Programming 1.6 Object Oriented Programming languages 1.7 Structure of c++ Program 1.8 Output using cout 1.9 Directives:-pre-processor directives, header files, The using Directives, Comments 1.10 Basic Data types 1.11 Input using cin 1.12 Overview of operators: Types of operators, scope-resolution operator 1.13 Manipulators and Enumeration
Unit– 2: Function ,Structure and Working with Object	2a. Develop program using call by reference and return by reference, default arguments, constant arguments, inline and function overloading. 2b. Develop program using structure. 2c. Apply concept of access specifier in C++ 2c. Develop Simple Programs using class and objects, array of objects, friend functions, passing and	**Function** 2.1 Introduction to Function 2.2 call by value and call by reference 2.3 Returning values from function 2.4 Overloaded function:-different number of arguments, different kinds of arguments 2.5 Inline function 2.6 Default arguments **Structure** 2.7 A simple Structure 2.8 Defining structure 2.9 Defining structure variable 2.10 Accessing structure members 2.11 Initializing structure member **Class and Object**

	returning objects and friend class 2d. Apply concept of static member and static member function in C++	2.12 Introduction to class and object 2.13 Declaration of class and object 2.14 Access Specifier-Private, public and protected 2.15 Defining member function inside 2.16 Defining member function outside of the class using scope resolution operator 2.17 private member function 2.18 outside member function as inline 2.19 static member and member function 2.20 array of object 2.21 object as a function argument 2.22 friend function 2.23 returning object 2.24 friend class
Unit– 3: Constructor and Destructor	3a. Define constructor & destructor 3b. Develop program using constructor and destructor	3.1 Introduction to constructor and Destructor 3.2 characteristics of constructor and destructor 3.3 Define and initialize constructor 3.4 constructor with arguments(parameterized constructor) 3.5 overloading constructor(multiple constructors) 3.6 array of object using constructors 3.7 constructor with default arguments 3.8 copy constructor 3.9: destructor
Unit–4: Inheritance	4a. Define Inheritance 4b. List the applications of inheritance, types of inheritance and develop programs using single, multilevel and multiple inheritance 4. Apply the concept of constructor in derived classes	4. Introduction to Inheritance and Reusability 4. Simple Inheritance: Using public, Private and protected derivation 4.3 Protected Data with private and public inheritance 4.4 Single Inheritance 4.5 Multilevel Inheritance 4.6 Multiple Inheritance 4.7 Hierarchical Inheritance 4.8 Hybrid Inheritance 4.9 Multipath Inheritance 4.10 Virtual Base Class

		4.11 Constructors in base and derive classes 4.12 abstract class 4.13 advantages and disadvantages of inheritance
Unit–5: MS- Polymorphism, Virtual Function and working with Files	5a. Apply this Pointer to Objects 5b. Develop a program using runtime polymorphism. 5c. Develop a program using File operations.	5.1 this pointer 5.2 Compile time and runtime polymorphism 5.3 virtual function and pure virtual function 5.4 Introduction to File Stream Classes, 5.5 Opening and closing a file- File opening modes.

***Note**: The UOs need to be formulated at the 'Application Level' and above of Revised Bloom's Taxonomy' to accelerate the attainment of the COs and the competency.*

9. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total Marks
I	Principles of Object Oriented Programming	06	4	4	2	10
II	Function ,Structure and Working with Object	14	2	10	12	24
III	Constructor and Destructor	08	2	6	6	14
IV	Inheritance	10	2	6	8	16
V	Polymorphism, Virtual Function and working with Files	04	2	2	2	6
Total		**42**	**12**	**28**	**30**	**70**

***Legends:** R=Remember, U=Understand, A=Apply and above (Revised Bloom's taxonomy)*

***Note**: This specification table provides general guidelines to assist student for their learning and to teachers to teach and question paper designers/setters to formulate test items/questions assess the attainment of the UOs. The actual distribution of marks at different taxonomy levels (of R, U and A) in the question paper may vary from above table.*

10. SUGGESTED STUDENT ACTIVITIES

Other than the classroom and laboratory learning, following are the suggested student-related ***co-curricular*** activities which can be undertaken to accelerate the attainment of the various outcomes in this course: Students should conduct following activities in group and prepare small reports (of 1 to 5 pages for each activity). For micro project report should be as per suggested format, for other activities students and teachers together can decide the format of the report. Students should also collect/record physical evidences such as photographs/videos of the activities for their (student's) portfolio which will be useful for their placement interviews:

a) Students are encouraged to learn Visual Language programming like scratch, snap etc.
b) Undertake micro-projects in teams.
c) Prepare charts to explain use/process of the identified topic.
d) https://www.codechef.com/ , in this website very elementary programs are available, students are expected to solve those programs
e) https://code.org/, an hour of code may be organized and students are encouraged to participate
f) Students are encouraged to register themselves in various MOOCs such as: Swayam, edx, Coursera, Udemy etc to further enhance their learning.
g) List the applications which are developed using C
h) Encourage students to participate in different coding competitions like hackathon, online competitions on codechef etc.
i) Encourage students to form a coding club at institute level and can help the slow learners

11. SUGGESTED SPECIAL INSTRUCTIONAL STRATEGIES (if any)

These are sample strategies, which the teacher can use to accelerate the attainment of the various outcomes in this course:

a) Massive open online courses (***MOOCs***) may be used to teach various topics/sub topics.
b) Guide student(s) in undertaking micro-projects.
c) Managing Learning Environment
d) Diagnosing Essential Missed Learning concepts that will help for students.
e) Guide Students to do Personalized learning so that students can understand the course material at his or her pace.
f) Encourage students to do Group learning by sharing so that teaching can easily be enhanced.
g) ***'CI" in section No. 4***means different types of teaching methods that are to be employed by teachers to develop the outcomes.
h) About ***20% of the topics/sub-topics*** which are relatively simpler or descriptive in nature is to be given to the students for ***self-learning***, but to be assessed using different assessment methods.
i) With respect to ***section No.11***, teachers need to ensure to create opportunities and provisions for ***co-curricular activities***.
j) Guide students on how to address issues on environment and sustainability using the knowledge of this course

12. SUGGESTED MICRO-PROJECTS

Only one micro-project is planned to be undertaken by a student that needs to be assigned to him/her in the beginning of the semester. In the first four semesters, the micro-project are group-based (group of 3 to 5). However, **in the fifth and sixth semesters**, the number of students in the group should ***not exceed three.***

The micro-project could be industry application based, internet-based, workshop-based, laboratory-based or field-based. Each micro-project should encompass two or more COs which are in fact, an integration of PrOs, UOs and ADOs. Each student will have to maintain dated work diary consisting of individual contribution in the project work and give a seminar presentation of it before submission. The total work load on each student due to the micro-project should be about ***16 (sixteen) student engagement hours*** (i.e., about one hour per week) during the course. The students ought to submit micro-project by the end of the semester (so that they develop the industry-oriented COs).

A suggestive list of micro-projects is given here. This should relate highly with competency of the course and the COs. Similar micro-projects could be added by the concerned course teacher:

a) Develop C++ program for Library Management System. In this user can enter the record of new books and retrieve the details of books available in the library. User can issue the books to the students and maintain their records. Late fine is charged for students who return the issued books after the due date. Only one book is issued to students. New book is not issued to students those not returned the last
b) Develop C++ Program for Banking Record system. In this user can add, edit, search, delete or adjust records in files of Bank.
c) Develop C++ Program for Bus Reservation system. In this user can add bus information, reserve bus seat, display reservation information, and receive information about buses that are available.
d) Develop C++ Program for Phone Book Management system. In this user can adding, searching, modifying, listing, and deleting records through the use of file.
e) Develop C++ Program for Student Database Management system. In this user can adding, searching, modifying, listing, and deleting records through the use of file.
f) Develop C++ Program for Telephone Directory system. In this user can adding, searching, modifying, listing, and deleting records through the use of file.

13. SUGGESTED LEARNING RESOURCES

S. No.	Title of Book	Author	Publication with place, year and ISBN
1	Object Oriented Programming in C++	Lafore, Robert	SAMS, 2012
2	Object Oriented Programming with C++	Balagurusamy, E.	McGrawHill, Delhi, 2012
3	Object Oriented Programming with C++ - second edition	Sahay, Sourav	Oxford, Delhi 2012
4	Mastering C++	Venugopal	Tata McGrawHill, Delhi,2011
5	Programming in c++	Kamthane, Ashok	Pearson, New Delhi, 2012
6	C++ An Introduction to Programming	Jesse Liberty, Jim Keogh	Prentice-Hall, India
7	The Complete Reference C++	Herbert Schildt	Tata McGraw-Hill

14. SUGGESTED LEARNING WEBSITES

a) https://snap.berkeley.edu/snap/snap.html
b) https://scratch.mit.edu/download/scratch2
c) https://nptel.ac.in/courses/106/105/106105151/
d) https://www.programiz.com/cpp-programming
e) https://www.codecademy.com/learn/learn-c-plus-plus
f) https://www.tutorialspoint.com
g) www.w3schools com
h) https://www.udemy.com/topic/c-plus-plus/
i) https://www.udacity.com/course/c-for-programmers--ud210

15. PO-COMPETENCY-CO MAPPING

Semester II	Basic Object Oriented Programming (Course Code:4320702)									
	POs and PSOs									
Competency & Course Outcomes	PO 1 Basic & Discipline specific knowledge	PO 2 Problem Analysis	PO 3 Design/ development of solutions	PO 4 Engineering Tools, Experimentation &Testing	PO 5 Engineering practices for society, sustainability & environment	PO 6 Project Management	PO 7 Life-long learning	PSO 1	PSO 2	PSO 3 (If needed)
Competency ***Use Fundamentals of Computer in various engineering applications***										
Course Outcomes CO a) Select procedural oriented and object oriented approach to solve given problem	3	2	2	2	-	-	1			
CO b) Implement object-oriented program using constructor and destructor	2	2	2	2	-	-	1			
CO c) Implement Inheritance for code reuse in c++ program	2	2	2	2	-	-	1			
CO d) Develop program using runtime polymorphism	2	2	2	2	-	-	1			

Legend: '**3**' for high, '**2**' for medium, **'1'** for low or **'-'** for the relevant correlation of each competency, CO, with PO/ PSO

16. COURSE CURRICULUM DEVELOPMENT COMMITTEE

GTU Resource Persons

S. No.	Name and Designation	Institute	Contact No.	Email
1	Shri P. P. Kotak Principal	Government Polytechnic, Rajkot	9825469617	Kotakp2003@yahoo.com
2	Smt. M. P. Mehta HOD	Government Polytechnic, Gandhinagar	9879578273	manishamehtain@gmail.com
3	Smt. Jigna N.Acharya	K.D.Polytechnic, Patan	9428752038	jignaforever@gmail.com