Yogoda Satsanga Mahavidyalaya
JAGANNATHPUR, DHURWA, RANCHI – 834004
Email address: ysmranchi4@gmail.com
(NAAC Accredited, Grade: B++, CGPA: 2.89)

COURSE PLAN

NAME OF THE DEPARTMENT: **BCA/IT**

NAME OF THE FACULTY: **Prof. Saroj Kumari, Prof. Partha Sarathi Chattaraj, Prof. Abhishek Kumar Vishwakarma, Prof. Khushbu Kumari**

ACADEMIC SESSION: **2023-24**

YEAR: **2024**

PROGRAMME: **BCA & B.Sc. (IT)**

SEMESTER: **I**

COURSE TYPE: **BCA/IT**

COURSE NAME: **C AND C++**

COURSE CODE: **C1**

TOTAL CREDIT: **6**

PROGRAMME OUTCOMES (PO):

PO1: Scientific & Computational Knowledge: - Apply the information on scientific & computational ideas, software engineering and innovation basics.

PO2: Problem Analysis, Design & Implementation: - Identify, formulate and analyze real world problem. Design solution for Software, Hardware & Networking problems and implementation using Software & Network tools.

PO3: Modern tool usage: - Ability to select modern computing tools, skills and techniques necessary for innovative software solutions.

PO4: Project Management: -Comprehend Software Engineering and Technology standards and apply these to prepare own project and system as a part and pioneer in a group.

PO5: Career Development & Entrepreneurship: Classify opportunities, private enterprise dream and use of original thoughts to build worth and means for the betterment of the human being and the world.

PO6: C*ommunication:* Communicate effectively on computational & information Technology 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.

PO7: Ethics: Ability to apply and commit professional Ethics, cyber regulations & control on software piracy in a global economic environment.

PO8: Preparing students for future aspects: Building and improving their creativity, social awareness, and general knowledge.

PO9: 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 changes.

PROGRAMME SPECIFIC OUTCOMES (PSO):

PSO1: An ability to apply technical comprehension in varied areas of Computer Applications and experience a conducive environment in cultivating skills for thriving career and higher studies.

PSO2: Understand the concept of Programing logic, Web designing logic, Signal processing, Image processing, Mobile Applications, Multimedia Media.

PSO3: Develop competencies in various disciplines of technologies such as Server-side Web applications, computer networking, software engineering, database concepts and programming

COURSE OUTCOMES (COs):

CO1: Learn the basic of procedural and object oriented programming, structure of C and C++ programming its compilation & execution.

CO2: Understand the concept of Data type, variables, Constants, Operators & basic of I/O Operations in C & C++.

CO3: To know the Expressions, Conditional Statements (section, jumping) and iterative statements in C & C++.

CO4: Learn, manipulating & implementation of user defined functions, built in functions, One Dimensional Arrays & Multiple Dimensional Arrays.

CO5: To know (declaring, initializing) & implementation of Derived Data Types: Structures and Unions using C and C++,

CO6: Learn (declaring, initializing) and Implementation of Pointers and References in C and C++.

CO7: Understand the concept of deducing the memory Allocation in C++, differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions.

CO8: Perceive File I/O, Preprocessor Directives, opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes in C++).

CO9: Understand Classes & objects, array of objects, Class Constructors, Constructor Overloading , function overloading & operator overloading (Unary & binary).

CO10: Apply & implement the concept of Inheritance, Polymorphism & Exception Handling.

COURSE TEACHING AND LEARNING ACTIVITIES

A. PEDAGOGY

i.	**Whiteboard**	√
ii.	**Flipped Class**	√
iii.	**PPT**	√

B. COURSE COMPLETION PLAN

UNIT	NO. OF LECTURES		TEST	QUIZ	ASSIGNMENT
	THEORY	PRACTICAL/TUTORIAL			
1	2	3	√	√	√
2	4	6	√	√	√
3	4	6	√	√	√
4	8	12	√	√	√
5	3	4	√	√	√
6	5	9	√	√	√
7	3	3	√	√	√
8	3	5	√	√	√
9	5	9	√	√	√
10	3	7	√	√	√
11	6	10	√	√	√

A. COURSE DELIVERY PLAN:

UNIT	TOPIC/SUBTOPIC	LECTURE REQUIRED (Theory & Practical)	CO ADDRESSED	ASSIGNMENT/ TEST/ QUIZ
1	Understand the basic terminology used in computer programming and simple programming in C/C++	5	CO1	√
2	Use different data types, Variables, Using Named Constants, Keywords, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf(), cin, cout), Using Basic Header Files (stdio.h, iostream.h, conio.hetc). in a computer program.	10	CO1	√

3	Design programs involving decision structures, Simple Expressions in C/C++ (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)	**10**	**CO2**	√
4	Explain the difference between call by value and call by reference. Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments. Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays	**20**	**CO2,CO3**	√
5	Use different data structures, utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.	**7**	**CO2,CO3**	√
6	Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring	**14**	**CO2,CO3, CO4**	√

	and initializing references, Using references as function arguments and function return values			
7	Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, use of new and delete operators, storage of variables in static and dynamic memory allocation	**6**	**CO3**	√
8	Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros	**8**	**CO5**	√
9	Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables &Functions, Objects as parameters, Specifying the Protected and Private Access, Copy Constructors,Overview of Template classes and their use.	**14**	**CO1,CO6**	√
10	Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators)	**10**	**CO6**	√
11	Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Basics Exceptional Handling (using catch and throw, multiple catch statements), Catching all exceptions, Restricting exceptions, Rethrowing exceptions.	**16**	**CO7**	√

B. COURSE OUTCOME ASSESSMENT PLAN

a. DIRECT ASSESSMENT

(Please tick the appropriate column)

COURSE OUTCOME	ASSESSMENT				REMARKS
	QUIZ	TEST	MID	END	

			SEMESTER	SEMESTER	
CO1			√	√	
CO2		√	√	√	
CO3			√	√	
CO4		√	√	√	
CO5			√	√	
CO6		√	√	√	
CO7			√	√	
CO8		√	√	√	
CO9			√	√	
CO10		√	√	√	

b. INDIRECT ASSESSMENT (STUDENT SURVEY)

Rate the following aspects of course outcomes. Use the scale 1-3

S. No	Course Outcome	1	2	3
1.	CO1			√
2.	CO2			√
3.	CO3		√	
4.	CO4			√
5.	CO5		√	
6.	CO6		√	
7.	CO7			√
8.	CO8			√
9.	CO9			√
10.	CO10			√

1. Average
2. Good
3. Very Good

C. SUGGESTED READINGS

a. TEXT BOOKS

Herbtz Schildt, "C++: The Complete Reference", Fourth Edition, McGraw Hill.2003

b. REFERENCE BOOKS

- Bjarne Stroustrup, "The C++ Programming Language", 4th Edition, Addison-Wesley, 2013.
- E Balaguruswamy, "Object Oriented Programming with C++"
- Robert Lofore, "Object Oriented Programming with C++"

c. VIDEO RESOURCE

- https://www.youtube.com/watch?v=AGrcyWV7hL8&list=PLrjkTql3jnm-Voi7giH4JITCi6cuZSN42
- https://www.youtube.com/watch?v=j8nAHeVKL08&list=PLu0W_9lII9agpFUAlPFe_VNSlXW5uE0YL

d. WEB RESOURCES:

- https://www.programiz.com/cpp-programming
- https://www.javatpoint.com/cpp-tutorial
- https://www.tutorialspoint.com/cplusplus/index.htm

e. E-RESOURCES

- Notes in the form of PDF share to the Students WhatsApp group.