



LEARN



IN JUST 30 DAYS

DAY 1

Introduction to C++



TOPICS

- What is C++?
- Setting up a development environment (e.g., using an IDE like Visual Studio or an online compiler).
- Writing your first "Hello, World!" program.



RESOURCES

1. https://www.w3schools.com/cpp/cpp_getstarted.asp
2. https://www.onlinegdb.com/online_c++_compiler
3. <https://www.geeksforgeeks.org/setting-c-development-environment/>



? QUESTIONS

1. What is the difference between C and C++?
2. Explain the compilation process in C++.
3. How do you execute a C++ program and what happens behind the scenes?



DAY 2

Variables and Literals



TOPICS

- Declaring and initialising variables.
- Different data types (int, double, char, etc.).
- Constants and literals.



RESOURCES

1. <https://www.programiz.com/cpp-programming/variables-literals>
2. <https://www.geeksforgeeks.org/literals-in-c-cpp-with-examples/>
3. https://www.w3schools.com/cpp/cpp_variables.asp



? QUESTIONS

1. Declare an integer variable named "age" and initialize it with your age.
2. Create a constant called "PI" and set its value to 3.14159.
3. Calculate the area of a circle with a radius of 5 units.



DAY 3

Basic I/O



TOPICS

- Input using cin and output using cout.
- Input and Output Formatting techniques



RESOURCES

1. <https://www.geeksforgeeks.org/basic-input-output-c/>
2. https://www.tutorialspoint.com/cplusplus/cpp_basic_input_output.htm
3. https://cplusplus.com/doc/tutorial/basic_io/



? QUESTIONS

1. Write a program that takes two numbers as input and prints their sum.
2. Format the output of a temperature in Celsius and Fahrenheit side by side.
3. Write a program to take two floating point numbers as input and perform different arithmetic operations on them
4. What are the different formatting characters used in C++?



DAY 4

Control Flow (if...else and switch statement)



TOPICS

- Conditional statements and boolean expressions.
- Using if, else if, and else.



RESOURCES

1. <https://www.programiz.com/cpp-programming/if-else>
2. [https://www.w3schools.com/cpp/
cpp_conditions_elseif.asp](https://www.w3schools.com/cpp/cpp_conditions_elseif.asp)



? QUESTIONS

1. Write a program that checks if a given number is positive, negative, or zero.
2. Implement a program that determines if a student's grade is "A," "B," "C," "D," or "F" based on their score.
3. Create a program that determines the day of the week based on the user's input (1 for Monday, 2 for Tuesday, etc.).
4. Implement a simple calculator program that supports addition, subtraction, multiplication, and division using a switch statement.



DAY 5

Control Flow (for Loop, while Loop and do...while Loop)



TOPICS

- Introduction to loops.
- Using for loops for iteration.
- Using while and do...while loops.
- Loop control statements (break and continue).



RESOURCES

1. <https://www.javatpoint.com/cpp-flow-control>
2. <https://cplusplus.com/doc/tutorial/control/>
3. <https://subscription.packtpub.com/book/programming/9781789801491/1/ch01lvl1sec06/control-flow-statements>

? QUESTIONS

1. Create a program that prints all even numbers from 1 to 20.
2. Calculate the sum of the first 100 natural numbers using a for loop.
3. Write a program to find the factorial of a number using a do...while loop.



DAY 6

Functions



TOPICS

- Defining functions, parameters, and return types.



RESOURCES

1. <https://www.geeksforgeeks.org/functions-in-cpp/>
2. https://www.w3schools.com/cpp/cpp_functions.asp

? QUESTIONS

1. Write a function that calculates the square of a number and returns the result.
2. Create a program that uses a function to check if a given year is a leap year.
3. Write a function that takes a number and prints a multiplication table of the number.



DAY 7

Function Overloading and Default Arguments



TOPICS

- Overloading functions with different parameters.
- Default argument values.
- Hands-on: Overload your max function to work with three numbers and test it.



RESOURCES

1. <https://www.geeksforgeeks.org/function-overloading-c/>
2. <https://www.javatpoint.com/cpp-overloading>
3. <https://www.geeksforgeeks.org/default-arguments-c/>
4. https://en.cppreference.com/w/cpp/language/default_arguments



? QUESTIONS

1. Overload a function to find the maximum of three numbers.
2. Modify the max function to have default arguments, allowing it to work with two or three numbers.
3. Explain the concepts of function overloading and default arguments with examples.



DAY 8-9

Storage Classes and Recursion



TOPICS

- Storage classes (auto, static, extern).
- Introduction to recursion in C++



RESOURCES

1. <https://www.geeksforgeeks.org/storage-classes-in-c-with-examples/>
2. <https://www.javatpoint.com/cpp-storage-classes>
3. https://www.w3schools.com/cpp/cpp_functions_recursion.asp
4. <https://www.geeksforgeeks.org/cpp-recursion/>



? QUESTIONS

1. Write a recursive function to calculate Fibonacci numbers.
2. Declare a global variable and a static variable inside a function. Compare their lifetimes.
3. Write a recursive function to calculate the factorial of a number.
4. Discuss the storage classes in C++ (auto, static, extern, etc.) and their use cases.



DAY 10

Arrays



TOPICS

- Introduction to arrays.
- Array initialization and access.



RESOURCES

1. https://www.w3schools.com/cpp/cpp_arrays.asp
2. <https://www.geeksforgeeks.org/cpp-arrays/>
3. <https://cplusplus.com/doc/tutorial/arrays/>



? QUESTIONS

1. Describe the concept of arrays in C++.
+. How are they different from other data structures?
2. Write a program to find the sum and average of elements in an array
3. Discuss advantages and limitations of using arrays.
4. Create a program that finds the largest and smallest elements in an array and their positions.



DAY 11

Multidimensional Arrays



TOPICS

- Creating and using multidimensional arrays.



RESOURCES

1. https://www.w3schools.com/cpp/cpp_arrays_multi.asp
2. <https://www.programiz.com/cpp-programming/multidimensional-arrays>



? QUESTIONS

1. Implement a program to perform matrix multiplication.
2. Write a program to perform matrix subtraction and addition.
3. Describe the use cases for multidimensional arrays and compare them to one-dimensional arrays.
4. How is a multidimensional array stored in memory in C++?



DAY 12

Strings



TOPICS

- Introduction to C++ strings.
- String manipulation functions.



RESOURCES

1. <https://www.geeksforgeeks.org/stdstring-class-in-c/>
2. https://www.tutorialspoint.com/cplusplus/cpp_strings.htm
3. <https://cplusplus.com/reference/string/string/>



? QUESTIONS

1. Create a program to reverse a given string.
2. Discuss the dynamic nature of C++ strings and the benefits of using them over fixed-size character arrays.
3. Given a string and a substring write a C++ program to find the location of the substring
4. Develop a program that checks if a given string is a palindrome.



DAY 13

Structures and Enums



TOPICS

- Defining structures and enums in C++



RESOURCES

1. https://www.w3schools.com/cpp/cpp_structs.asp
2. <https://www.programiz.com/cpp-programming/enumeration>
3. <https://www.geeksforgeeks.org/enumeration-in-cpp/>
4. <https://www.geeksforgeeks.org/structures-in-cpp/>



? QUESTIONS

1. Design a structure for storing information about books.
2. What is the purpose of structures and enumerations in C++?
3. Define a structure for storing information about students and create an array of student records.
4. Define a structure representing a point in 3D space and implement functions to calculate distance and midpoint between two points.



DAY 14-15

Introduction to Classes



TOPICS

- What is object oriented programming in C++?
- Declaring and defining classes.



RESOURCES

1. https://www.w3schools.com/cpp/cpp_classes.asp
2. <https://www.geeksforgeeks.org/c-classes-and-objects/>
3. <https://cplusplus.com/doc/tutorial/classes/>
4. <https://www.geeksforgeeks.org/object-oriented-programming-in-cpp/>



? QUESTIONS

1. Create a simple class to represent a car.
2. Define a class in C++. What are its components?
3. Create a class representing a "Person" with attributes like name and age, and instantiate objects of this class.
4. Describe the four pillars of object-oriented programming (OOP) and how C++ supports each of them.



DAY 16

Constructors and Destructors



TOPICS

- Constructors and destructors in classes.
- Parameterized Constructors



RESOURCES

1. <https://www.geeksforgeeks.org/constructors-c/>
2. <https://en.cppreference.com/w/cpp/language/constructor>
3. <https://www.programiz.com/cpp-programming/constructors>



? QUESTIONS

1. Add constructors to your car class and create car objects.
2. Explain the role of constructors and destructors in classes. What are copy constructors?
3. Create a class for a bank account with attributes for balance and account number, and implement methods for deposit and withdrawal. Add parameterized constructors and a destructor to your bank account class.



Objects and Member Functions



TOPICS

- Creating objects of a class.
- Accessing member functions and variables.



RESOURCES

1. https://www.tutorialspoint.com/cplusplus/cpp_class_member_functions.htm
2. <https://www.ibm.com/docs/en/zos/2.1.0?topic=only-member-functions>
3. <https://www.javatpoint.com/cpp-class-member-functions>

? QUESTIONS

1. Implement methods in your car class to set and display car details.
2. Describe how objects interact with member functions in C++ classes.
3. Discuss the concept of encapsulation in OOP and how it relates to class objects and member functions.
4. Create multiple bank account objects and demonstrate encapsulation by accessing and modifying their attributes.



DAY 18

Operator Overloading



TOPICS

- Overloading operators like +, -, *, etc in C++



RESOURCES

1. <https://www.geeksforgeeks.org/operator-overloading-cpp/>
2. <https://www.programiz.com/cpp-programming/operator-overloading>
3. https://www.tutorialspoint.com/cplusplus/cpp_overloading.htm



? QUESTIONS

1. Create a class to represent complex numbers. Overload the + operator in your car class to add two complex numbers.
2. What is the significance of operator overloading in C++? Provide examples of commonly overloaded operators.
3. Overload the "+" operator to concatenate two "Person" objects' names.



DAY 19-20

Pointers and Dynamic Memory Allocation



TOPICS

- Introduction to pointers.
- Dynamic memory allocation with new and delete.



RESOURCES

1. <https://dev.to/denvercoder1/c-pointers-and-dynamic-memory-allocation-1emi>
2. <https://www.geeksforgeeks.org/new-and-delete-operators-in-cpp-for-dynamic-memory/>
3. <https://cplusplus.com/doc/tutorial/dynamic/>



? QUESTIONS

1. Explain the concept of pointers and dynamic memory allocation in C++.
2. Create a dynamic array of "Person" objects and manipulate them using pointers.
3. Describe the advantages and potential pitfalls of using pointers in C++ for dynamic memory allocation.



DAY 21

Pointers and Functions



TOPICS

- Passing pointers as function parameters.



RESOURCES

1. <https://www.geeksforgeeks.org/function-pointer-in-cpp/>
2. <https://www.geeksforgeeks.org/passing-pointers-to-functions-in-cpp/>
3. https://www.tutorialspoint.com/cplusplus/cpp_passing_pointers_to_functions.htm



? QUESTIONS

1. Write a function to sort an array of cars using pointers.
2. How can pointers be passed as function parameters in C++?
3. Write a function that takes a pointer to an array and returns the sum of its elements.
4. Explain the concept of passing pointers as function parameters and how it affects the original data.



DAY 22

Inheritance



TOPICS

- Introduction to inheritance.
- Base and derived classes.



RESOURCES

1. <https://www.geeksforgeeks.org/inheritance-in-c/>
2. https://www.w3schools.com/cpp/cpp_inheritance.asp
3. <https://www.programiz.com/cpp-programming/inheritance>



? QUESTIONS

1. Create a base class "Vehicle" and a derived class "Car."
2. Create a base class "Shape" and derived classes "Circle" and "Rectangle."
3. Describe the benefits and challenges of inheritance in OOP, and how C++ addresses those challenges.
4. Explain the concept of inheritance and its advantages in object-oriented programming.



DAY 23

Access Control in Inheritance



TOPICS

- Public, private, and protected access specifiers.



RESOURCES

1. <https://www.programiz.com/cpp-programming/public-protected-private-inheritance>
2. <https://www.learncpp.com/cpp-tutorial/inheritance-and-access-specifiers/>
3. <https://www.geeksforgeeks.org/cpp-inheritance-access/>



? QUESTIONS

1. Discuss the access control modifiers in C++ inheritance (public, private, protected).
2. Modify your "Vehicle" hierarchy to demonstrate the accessibility of base class members in derived classes.



DAY 24

Function Overriding and Overloading



TOPICS

- Overriding base class functions in derived classes.
- Overloading of functions in C++



RESOURCES

1. <https://www.geeksforgeeks.org/function-overloading-vs-function-overriding-in-cpp/>
2. <https://www.geeksforgeeks.org/function-overriding-in-cpp/>
3. <https://www.programiz.com/cpp-programming/function-overloading>



? QUESTIONS

1. Define function overriding and its significance in polymorphism.
Provide examples.
2. What is the difference between function overloading and overriding?
3. Override a member function in your "Circle" and "Rectangle" classes to calculate area.



DAY 25

Multiple Inheritance and Virtual Functions



TOPICS

- Multiple and multilevel inheritance.
- Virtual functions and dynamic binding.



RESOURCES

1. <https://isocpp.org/wiki/faq/multiple-inheritance>
2. <https://www.geeksforgeeks.org/multiple-inheritance-in-c/>
3. <https://www.geeksforgeeks.org/multiple-inheritance-in-c/>
4. <https://www.javatpoint.com/cpp-virtual-function>



? QUESTIONS

1. Explain multiple inheritance and virtual functions in C++.
2. Discuss multiple inheritance and the "diamond problem." How does C++ handle this issue?
3. Implement multiple inheritance to derive a 3 dimensional shape using length, breadth and height.
4. Create a class hierarchy involving multiple inheritance and use virtual functions to achieve dynamic binding.



DAY 26

Friend Functions and Classes



TOPICS

- Understanding friend classes
- Implementing Friend functions



RESOURCES

1. <https://www.geeksforgeeks.org/friend-class-function-cpp/>
2. <https://www.programiz.com/cpp-programming/friend-function-class>
3. <https://www.javatpoint.com/cpp-friend-function>



? QUESTIONS

1. What is a friend function, and how does it access class members?
2. Implement a friend function to calculate the average speed of a vehicle based on distance and time.
3. Define a friend function for your "Shape" classes to display their details.



DAY 27

C++ Templates



TOPICS

- Introduction to class templates.
- Introduction to function templates



RESOURCES

1. <https://www.geeksforgeeks.org/templates-cpp/>
2. <https://www.programiz.com/cpp-programming/class-templates>
3. https://www.tutorialspoint.com/cplusplus/cpp_templates.htm



? QUESTIONS

1. Describe class templates in C++.
How do they enable generic programming?
2. Create a template class for a generic array and use it with different data types.
3. Explain how to make a function template with an example.



Final Review and Practice

1. LIBRARY CATALOG

- Build a library catalog system where you can add, remove, and search for books.
- Use classes and structures to represent books and manage collections.
- Employ arrays or linked lists for book storage.

2. SIMPLE GAME (e.g., Guess the Number)

- Create an interactive game where the computer generates a random number, and the player tries to guess it.
- Use loops (for or while), conditional statements, and functions for game logic.
- Keep track of the player's score.



WHY BOSSCODER?

 **1000+** Alumni placed at Top Product-based companies.

 More than **136% hike** for every **2 out of 3** working professional.

 Average package of **24LPA**.

The syllabus is most up-to-date and the list of problems provided covers all important topics.

Lavanya
 Meta



Course is very well structured and streamlined to crack any MAANG company

Rahul .
 Google



[EXPLORE MORE](#)