

VERSION

2

CRASH COURSE

ELYSIUM
ACADEMY
CORE C & C++
PROGRAMMING

**ELYSIUM
ACADEMY**

**CORE C & C++
PROGRAMMING**

ELYSIUM
ACADEMY
CORE C & C++
PROGRAMMING

ELYSIUM
ACADEMY

MASTER

SR. CODE

EAPL/CRASH/CRTC01

COURSE CODE

EACCC

SUB CATEGORY

SOFTWARE DEVELOPMENT



TOTAL DURATION
45
HOURS



THEORY TAKEN
11
HOURS



PRACTICAL TAKEN
34
HOURS

COURSE DESCRIPTION



C and C++ Anybody interested in programming who wants to begin a career in the software business must learn programming. It is the fundamental building block of all programming languages. All programming languages, including Java, C#, and all object-oriented languages, are descended from C and C++. You will study all the fundamentals of C and C++ in this course from start before moving on to more complex topics. You will learn the course by practising the code as you progress through it. Each and every topic in this course is taught with example programmes to help you better comprehend the ideas.

COURSE GOALS



One of the major design goals for C++ was to support object-oriented programming. The strong compile-time typing of C++ adds complexity not found in pure object-oriented languages such as Smalltalk. The need to retain the efficiency and speed of C also influenced the support for OO programming.

FUTURE SCOPE



Since it is so flexible, it continues to be in great demand among professionals like software developers, game developers, C++ analysts, and backend developers, among others. According to the TIOBE ranking for 2022, C++ is the fourth most widely used language in the world.

01

CHAPTER

GETTING STARTED C

01. Beginning C

- a. What is C?
- b. Features of C
- c. Applications of C
- d. Structure of C
- e. Installation setup

02. Basic Terminologies in C

- a. Print statement
- b. Data types
- c. Variables
- d. Operators
- e. Identifiers
- f. Comments
- g. Escape Sequence
- h. Constants
- i. Tokens
- j. Keywords
- k. Hands on

03. Operators

- a. Types of operators
- b. Purpose of operators
- c. Arithmetic operators
- d. List the arithmetic operators
- e. Hands on arithmetic operators
- f. Increment and decrement operators
- g. Hands on increment and decrement operators



01
HRS
10
MINS



04
HRS
40
MINS

O4. Assignment and Relational operators

- a. List the assignment operators
- b. Purpose of using assignment operators
- c. Hands on assignment operators
- d. List the relational operators
- e. Purpose of using relational operators
- f. Hands on relational operators

O5. Logical and bitwise operators

- a. List the Logical operators
- b. Purpose of using Logical operators
- c. Hands on Logical operators
- d. List the bitwise operators
- e. Purpose of using bitwise operators
- f. Hands on bitwise operators
- g. List the other common operators
- h. Quiz on operators

02

CHAPTER

CONDITIONS & LOOP STATEMENTS

O1. Conditions

- a. If statement
- b. Else if Statement
- c. If else Statement
- d. Switch case Statement
- e. To check number is even or odd
- f. To check if a given number is a multiple of 3
- g. To check the character entered is vowel or not



01
HRS
00
MINS



03
HRS
20
MINS

O2. Loop Statements

- a. About Loops
- b. Why we used loops?
- c. Unconditional Branching using goto statement
- d. For loop
- e. To Print star patterns
- f. While loop
- g. do while loop
- h. break and continue
- i. Nested loop

03

CHAPTER

FUNCTIONS, ARRAY

O1. Functions in C:

- a. What is functions?
- b. Local and global variables
- c. Parameters and arguments
- d. Declaration and definition
- e. Storage class
- f. Pass by value and reference



01
HRS
30
MINS



03
HRS
20
MINS

- g. Recursive functions
- h. Functions like macros
- i. Math functions
- j. Hands on functions

O2. Recursion Function:

- a. What is Recursion function?
- b. Purpose of recursion
- c. Hands on recursion

O3. Arrays:

- a. What is Array?
- b. Declaration and Initialization
- c. Modify array
- d. Find length of the array
- e. Find minimum and maximum number in array
- f. Hands on declaration and initialization
- g. Create 2D array
- h. Find sum of numbers in 2D array

O4. Strings:

- a. What is string?
- b. String Concatenation
- c. Calculate String Length
- d. To compare two strings
- e. To reverse two strings
- f. To swap two strings
- g. Quiz

04

CHAPTER

POINTERS, DYNAMIC MEMORY ALLOCATION

O1. Pointers:

- a. What is pointers?
- b. Benefits of pointer
- c. Types of pointer
- d. Creating pointers
- e. Pointers and arrays
- f. How pointers are related to array?
- g. Pointer to a function
- h. Pointer to structure
- i. Pointer operators
- j. Pointer casting
- k. NULL Pointer
- l. The "sizeof" Operator
- m. Hands on 'sizeof'
- n. To swap two numbers using pointer

O2. Dynamic Memory Allocation:

- a. What is Dynamic Memory Allocation?
- b. Purpose of Dynamic Memory Allocation
- c. Difference between static memory allocation and dynamic memory allocation
- d. malloc()
- e. Hands on malloc()
- f. calloc()
- g. Hand on calloc()



01
HRS
20
MINS



02
HRS
40
MINS

- h. realloc()
- i. Hands on realloc()
- j. free()
- k. hands on free()

O3. Storage Classes and preprocessors

- a. What is storage classes?
- b. Types of storage classes
- c. Purpose of storage classes
- d. What is preprocessors?
- e. Macro Substitution
- f. File Inclusion

05

CHAPTER

STRUCTURES, FILE HANDLING

O1. Structures and Unions:

- a. Difference between structures and union
- b. What is structure?
- c. Structure Declaration
- d. Access structure members
- e. Strings in structure
- f. Hands on structures
- g. What is Union?
- h. Union Declaration
- i. Access union members
- j. Hands on union



01
HRS
30
MINS



03
HRS
20
MINS

O2. User defined types:

- a.enum
- b.Hands on enum
- c.Typedef
- d.Hands on Typedef

O3. File Handling in C:

- a.What is file handling?
- b.Why it is important?
- c.Types of files
- d.Creating a file
- e.Opening a file
- f. Reading a file
- g.Writing a file
- h.Deleting a file
- i. Functions of file handling
- j. Mode of operations

06

CHAPTER

GETTING STARTED WITH C++

O1. Beginning C++:

- a.What is C++?
- b.Why we used C++?
- c.How C++ differs from C
- d.Advantages and drawbacks
- e.Applications of C++
- f. Structure of C++
- g.Installation setup



01
HRS
20
MINS



04
HRS
20
MINS

O2. Data types and variables

- a. Primary data types
- b. Derived data types
- c. User defined data types
- d. Hands on data types
- e. Constants
- f. Variable declaration

O3. Programing structure

- a. Header file in C++
- b. What are header files?
- c. Name space
- d. Escape sequence
- e. Main Function
- f. Block and semicolon
- g. Key words

O4. Important Terminologies

- a. Function overloading
- b. Optional Parameters
- c. Reference Variables
- d. Basics of Console Input and Output
- e. Constant pointers
- f. Dynamic Memory allocation
- g. User input
- h. Problem understanding:
Sum of two numbers
- i. Problem understanding:
Average of three numbers

07

CHAPTER

FUNCTIONS

01. Develop programs

- a. Swap two numbers
- b. To print first and last no. in a three digit no.
- c. Find area and perimeter of a square
- d. Find area and perimeter of a rectangle
- e. Find quotient and remainder
- f. Reverse a number
- g. Multiply two numbers
- h. Create pyramid and patterns

02. Functions in C++

- a. Call by reference, Return by reference
- b. Function overloading and default arguments
- c. Inline function
- d. Static class members
- e. Friend functions
- f. Virtual Function
- g. Hands on functions in C++

03. Operator Overloading

- a. Overloading unary operations
- b. Overloading binary operators
- c. Data conversion
- d. pitfalls of operators overloading and conversion keywords
- e. Explicit and Mutable.



00
HRS
35
MINS



01
HRS
15
MINS

08

CHAPTER

OOPS

O1. OOPS

- a. What is OOPS?
- b. Purpose of OOPS
- c. Advantages of OOPS
- d. Application of OOPS
- e. Different category of OOPS

O2. Object and Class

- a. What is object?
- b. What is class?
- c. Create a class
- d. Create an object
- e. Class methods
- f. Types of class methods
- g. Data members
- h. this pointer
- i. static class member
- j. friend class and member
- k. Hands on object and class

O3. Constructors and Destructors

- a. What is constructors?
- b. Types of Constructors
- c. Default, Parameterized & Copy constructors
- d. Multiple constructors
- e. Constructors with default arguments & Dynamic constructor



01
HRS
50
MINS



04
HRS
40
MINS

f. What is destructors?

g. Hands on constructors and destructors

O4. Inheritance

a. What is inheritance?

b. Purpose of inheritance

c. Derived class

d. Members of derived class

e. Access Specifiers

f. Types of inheritance

g. Application of inheritance

O5. Types of inheritance

a. Single inheritance

b. Hands on single inheritance

c. Multiple inheritance

d. Hands on Multiple inheritance

e. Hierarchical inheritance

f. Hands on Hierarchical inheritance

g. Multilevel inheritance

h. Hands on Multilevel inheritance

i. Hybrid inheritance

j. Hands on Hybrid inheritance

09

CHAPTER

ENCAPSULATION, FILE HANDLING

O1. Encapsulation and polymorphism

- a. What is encapsulation
- b. Purpose of encapsulation
- c. Features of encapsulation
- d. Access private members
- e. Hands on encapsulation
- f. Polymorphism
- g. Why polymorphism?
- h. Run time polymorphism
- i. Compile time polymorphism
- j. Virtual Function
- k. Abstract Class
- l. Hands on polymorphism
- m. Quiz on OOPS

O2. I/O stream

- a. What is stream?
- b. Stream input
- c. Stream output
- d. Different Operations in I/O stream
- e. Basic syntax for different operations
- f. Hands on Operations in I/O stream



01
HRS
35
MINS



05
HRS
35
MINS

O3. File Handling

- a. What is file handling?
- b. File stream libraries
- c. Text file handling
- d. Binary file handling
- e. How to open files?
- f. Hands on open files
- g. How to close files?
- h. Hands on close files
- i. How to write to files?
- j. Hands on writing files
- k. How to read from files/
- l. Hands on reading files
- m. `getline()`

O4. Exception Handling

- a. What is Exception Handling?
- b. Purpose of exception handling
- c. Advantages and drawbacks
- d. try and catch block
- e. throw statement
- f. Hands on
- g. Quiz

O5. Templates

- a. About templates
- b. Function templates
- c. Class templates
- d. Create a class templates
- e. Hands on class templates
- f. Class template with multiple parameters
- g. Create a function templates
- h. Hands on function templates

Placement Assistance

100%

135+ Professional Courses

Practical Sessions

90%

67+ Global Pacts

Corporate Placements

65%

170+ IT Companies Tie-Up

ELYSIUM
GROUP OF
COMPANIES

**ELYSIUM
ACADEMY**

**PRIVATE
LIMITED**

AUTHORIZED INTERNATIONAL

Partners

