Mindluster Platform

# About the course OOPs in C

OOP stands for Object-Oriented Programming. Procedural programming is about writing procedures or functions that perform operations on the data, while object .

**Programming Category's Courses**

# Course Lesson(62)

Lesson 1 : **Object Oriented Programming | Introduction | Lec 1 | Bhanu Priya**

Lesson 2 : **Object and Class | OOPs in C | Lec 2 | Bhanu Priya**

Lesson 3 : **Abstraction and Encapsulation | OOPs in C | Lec 3 | Bhanu Priya**

Lesson 4 : **Inheritance and Polymorphism | OOPs in C | Lec 4 | Bhanu Priya**

Lesson 5 : **C programming introduction | Lec 5 | Bhanu Priya**

Lesson 6 : **difference between C and C | OOPs in C | Lec 6 | Bhanu Priya**

Lesson 7 : **Cin and Cout | Input Output Operators | OOPs in C | Lec 7 | Bhanu Priya**

Lesson 8 : **New and Delete Operator | OOPs in C | Lec 8 | Bhanu Priya**

Lesson 9 : **Encapsulation | OOPs in C | Lec 9 | Bhanu Priya**

Lesson 10 : **Data hiding | Comparision Encapsulation | OOPs in C | Lec 10 | Bhanu Priya**

Lesson 11 : **Abstract data type | ADT | Implementation | OOPs in C | Lec 11 | Bhanu Priya**

Lesson 12 : **Object State Behaviour Identity | OOPs in C | Lec 12 | Bhanu Priya**

Lesson 13 : **C Class declaration | Part 1 2 | OOPs in C | Lec 13 | Bhanu Priya**

Lesson 14 : **C Class declaration | Part 2 2 | OOPs in C | Lec 14 | Bhanu Priya**

Lesson 15 : **Abstract class | Example | Part 1 2 | OOPs in C | Lec 15 | Bhanu Priya**

Lesson 16 : **Abstract class | Example | Part 2 2 | OOPs in C | Lec 16 | Bhanu Priya**

Lesson 17 : **Constructor in C | Types | OOPs in C | Lec 17 | Bhanu Priya**

Lesson 18 : **Constructor in C | Calling types | OOPs in C | Lec 18 | Bhanu Priya**

Lesson 19 : **Destructor | Example | OOPs in C | Lec 19 | Bhanu Priya**

Lesson 20 : **Default parameters | OOPs in C | Lec 20 | Bhanu Priya**

Lesson 21 : **Dynamic memory allocation | Part 1 2 | OOPs in C | Lec 21 | Bhanu Priya**

Lesson 22 : **dynamic memory allocation | Part 2 2 | OOPs in C | Lec 22 | Bhanu Priya**

Lesson 23 : **Public and Private access specifiers | OOPs in C | Lec 23 | Bhanu Priya**

Lesson 24 : **Protected access specifier | OOPs in C | Lec 24 | Bhanu Priya**

Lesson 25 : **Inheritance | OOPs in C | Lec 25 | Bhanu Priya**

Lesson 26 : **Class hierarchy | OOPs in C | Lec 26 | Bhanu Priya**

Lesson 27 : **Single Inheritance | Multiple | Types | OOPs in C | Lec 27 | Bhanu Priya**

Lesson 28 : **Multilevel Inheritance | Types | OOPs in C | Lec 28 | Bhanu Priya**

Lesson 29 :
**Hierarchical and Hybrid Inheritance | Differences | Types | OOPs in C | Lec 29 | Bhanu Priya**

Lesson 30 : **Polymorphism | Introduction Types | OOPs in C | Lec 30 | Bhanu Priya**

Lesson 31 :
**Function Overloading Overriding | Static Polymorphism | OOPs in C | Lec 31 | Bhanu Priya**

Lesson 32 : **Operator Overloading in c | Static Polymorphism | OOPs in C | Lec 32 | Bhanu Priya**

Lesson 33 : **Virtual function in c | Dynamic Polymorphism | OOPs in C | Lec 33 | Bhanu Priya**

Lesson 34 : **Parametric Polymorphism | OOPs in C | Lec 34 | Bhanu Priya**

Lesson 35 : **Association | Examples | OOPs in C | Lec 35 | Bhanu Priya**

Lesson 36 : **Composition | Example program | OOPs in C | Lec 36 | Bhanu Priya**

Lesson 37 : **Aggregation | Example | OOPs in C | Lec 37 | Bhanu Priya**

Lesson 38 : **Function Template C | Generic | OOPs in C | Lec 38 | Bhanu Priya**

Lesson 39 : **Class Template in c | OOPs in C | Lec 39 | Bhanu Priya**

Lesson 40 : **Namespace in c | Part 1 2 | Rules | OOPs in C | Lec 40 | Bhanu Priya**

Lesson 41 : **Using namespace in C | Part 2 2 | OOPs in C | Lec 41 | Bhanu Priya**

Lesson 42 :
**Files and Streams | Part 1 3 | ifstream fstream ofstream | OOPs in C | Lec 42 | Bhanu Priya**

Lesson 43 :
**Files and Streams in c | Part 2 3 | Open Close Read Write | OOPs in C | Lec 43 | Bhanu Priya**

Lesson 44 : **Files and Streams in c | Part 3 3 | Examples | OOPs in C | Lec 44 | Bhanu Priya**

Lesson 45 : **Standard exceptions in c | OOPs in C | Lec 45 | Bhanu Priya**

Lesson 46 : **Exception handling in c | Part 1 3 | Introduction | OOPs in C | Lec 46 | Bhanu Priya**

Lesson 47 : **Exception Handling in c | Part 2 3 | Try Catch Throw | OOPs in C | Lec 47 | Bhanu Priya**

Lesson 48 :
**Exception handling in c | Part 3 3 | Multiple catch block | OOPs in C | Lec 48 | Bhanu Priya**

Lesson 49 : **Standard Template Library STL in c | Introduction | OOPs in C | Lec 49 | Bhanu Priya**

Lesson 50 : **STL Containers in c | Sequence Vector | Part 1 4 | OOPs in C | Lec 50 | Bhanu Priya**

Lesson 51 : **STL Containers in c | Sequence Deque List | Part 2 4 | OOPs in C | Lec 51 | Bhanu Priya**

Lesson 52 : **STL containers in c | Associative | Part 3 4 | OOPs in C | Lec 52 | Bhanu Priya**

Lesson 53 : **STL containers in c | Adapter | Part 4 4 | OOPs in C | Lec 53 | Bhanu Priya**

Lesson 54 : **STL Algorithms in c | Sorting | Part 1 4 | OOPs in C | Lec 54 | Bhanu Priya**

Lesson 55 : **STL Algorithms in c | Searching | Part 2 4 | OOPs in C | Lec 55 | Bhanu Priya**

Lesson 56 : **STL Algorithms in c | Non modifying | Part 3 4 | OOPs in C | Lec 56 | Bhanu Priya**

Lesson 57 : **STL Algorithms in c | Modifying Numeric | Part 4 4 | OOPs in C | Lec 57 | Bhanu Priya**

Lesson 58 : **Iterators in c | STL | Part 1 2 | OOPs in C | Lec 58 | Bhanu Priya**

Lesson 59 : **Iterators in c | STL | Part 2 2 | OOPs in C | Lec 59 | Bhanu Priya**

Lesson 60 : **Function objects in c | STL | OOPs in C | Lec 60 | Bhanu Priya**

Lesson 61 : **Manipulators in c | OOPs in C | Lec 61 | Bhanu Priya**

Lesson 62 : **Manipulators in c | User Defined | OOPs in C | Lec 62 | Bhanu Priya**

# Related courses

**React Projects Workshops**

**Node js back end JavaScript**

**Python programming language**

**Social Network Theme UI With Sass**

**Full Stack React Django**

**Build a Node js App With Sequelize**

for Business Contact
business@mindluster.com