

**CS304P LEAK PAPER FALL
2025**

Current paper solved paper

VIRTUAL EDUCATOR

<https://chat.whatsapp.com/JUjIXHi55aeCWpNvkSPdNj>

Virtual educator

CS304 (Object Oriented Programming)

Question No : 4 of 50

While overloading the stream extraction operator, the return type must be _____.

Answer

☐ Ostream

☐ Iostream

☒ Istream

Correct
Virtual Educator
03098851445

☐ Instream

CS304 (Object Oriented Programming)

Question No : 1 of 50

Vectors contain contiguous elements stored as a[an] _____.

Answer

☐

variable

☒

array

Correct
Virtual Educator
03098851445

☐

function

☐

datatype

CS304 (Object Oriented Programming)

Question No : 3 of 50

Which of the following is Not an example of error handling techniques?

Answer

- ☐ Abnormal Termination
- ☐ Graceful Termination
- ☐ Return the illegal value
- ☒ Returning a value

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 2 of 50

A range is often supplied to an algorithm by two _____ values.

Answer

- ☐ italic
- ☐ iteration
- ☒ iterator
- ☐ container

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 7 of 50

Which of the following is a mechanism of throwing the exception again after partial handling Input Iterators?

Answer

- ☐ Throw-Throw
- ☐ Throw-Again
- ☒ Re-Throw
- ☐ Partial-Throw

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 5 of 50

Suppose you create an uninitialized vector as follows:

```
vector<int> evec;  
After adding the statment,  
evec.push_back(21);  
what will happen?
```

Answer

- ☐ It will add an element to the start of evec and will initialize it with 21.
- ☐ It will add an element to the center of evec and will reinitialize it with 21.
- ☐ It will delete an element to the end (the back) of evec and will reinitialize it with 21.
- ☒ It will add an element to the end (the back) of evec and initialize it with 21.

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 15 of 50

A base class method (function) that is alternatively defined in derived class is known as _____.

Answer

- ☐ Overloaded
- ☐ Copy riding
- ☐ Overwriting
- ☒ Overriding

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 6 of 50

A container adapter is a constrained version of some _____

Answer

- ☒ First-class Container
- ☐ Second-class Container
- ☐ Third-class Container
- ☐ Fourth-class Container

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 16 of 50

A private data member of a Base class is not accessible in:

Answer

- ☐ Base class
- ☒ Derived class Correct
Virtual Educator
03098851445
- ☐ Friend class
- ☐ Both Base and Derived classes

CS304 (Object Oriented Programming)

Question No : 8 of 50

The operation of deep copy constructor must be defined by:

Answer

☐

System

☒

Programmer

CorrectVirtual Ed

☐

User

☐

Administrator

Question No : 14 of 50

In C++ while declaring a template class, there is no need to mention template _____ for class member functions.

Answer

- ☐ Parameter
- ☐ Return type
- ☐ Name
- ☒ Signature

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 13 of 50

If we write empty throw list for specifying exceptions then function will _____

Answer

- ☐ automatically throw exception
- ☒ not throw any exception
- ☐ throw default number of exceptions
- ☐ compiler will generate error

Correct
Virtual Educator
03098851445

CS304 (Object Oriented Programming)

Question No : 12 of 50

Which of the following is incorrect line regarding function template?

Answer

- ☐ `template<class T>`
- ☐ `template <typename U>`
- ☒ `class<template T>`
- ☐ `template < class T, class U>`

Correct
Virtual Educator
03098851445

When a virtual function is called by referencing a specific object by name and using the dot member selection operator (e.g., `squareObject.draw()`), the reference is resolved at _____

Answer

- ☐ Run time
- ☒ Compile time
- ☐ Processing time
- ☐ Execution time

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 15 of 30

Which of the following operator takes one argument while operator overloading as a member function of class?

Answer

☐ Both "+" and "*"

☐ "+"

☐ "*"

☒ "+="

Correct
Virtual Educator
03098851445

304P (Object Oriented Programming (Practical))

Question No : 17 of 30

When ____ operator is applied to a base class pointer, base class destructor is called regardless of the object type.

Answer

- ☐ Overloaded
- ☐ Assignment
- ☐ Dereference
- ☒ Delete

Correct
Virtual Educator
03098851445

Question No : 18 of 30

A mechanism that enables us to write single function or class that works for all data types is called ____ programming?

Answer

- ☐ Functional
- ☐ Procedural
- ☐ Structured
- ☒ Generic

Correct
Virtual Educator
03098851445

VIRTUAL

Question No : 16 of 30

A class with pure virtual function is a(an) _____ class and a class with no pure virtual function is a(an) _____ class.

Answer

☐ Pure Abstract, Pure Concrete

☐ Abstract, Friend

☐ Concrete, Abstract

☒ Abstract, Concrete

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 19 of 30

What must be true for the derived class to inherit from a template class?

Answer

- ☐ The derived class can have more template parameters than the base class.
- ☐ The derived class must have fewer template parameters than the base class.
- ☒ The derived class must take at least as many template parameters as the base class. ✓
- ☐ The derived class should not take any template parameters.

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 20 of 30

Which of the following is stream extraction operator in C++?

Answer

☐ <

☐ <<

☐ >

☒ >>

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 21 of 30

Which of the following is correct declaration of overloaded post increment operator for Test class?

Answer

☒ Test operator ++(int);

Correct
Virtual Educator
03098851445

☐ Test operator ++();

☐ Test ++ operator ();

☐ Test ++ operator (int);

CS304P (Object Oriented Programming (Practical))

Question No : 14 of 30

Which of the following is not a type of inheritance in C++?

Answer

☐ protected

☐ public

☐ private

☒ access

Correct
Virtual Educator
03098851445

Question No : 23 of 30

In case of protected inheritance between base class and derived class, private members of base class will become _____ in derived class.

Answer

☐ Protected

☒ Hidden

☐ Public

☐ Private

Correct
Virtual Educator
03098851445

VIRTUAL

CS304P (Object Oriented Programming (Practical))

Question No : 22 of 30

In Object Oriented Programming, _____ represents "IS A" relationship.

Answer

- ☐ Aggregation
- ☐ Friend
- ☒ Inheritance
- ☐ Composition

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 13 of 30

Identify the type of relationship between classes from the following class diagram.



Answer



Aggregation

Correct
Virtual Educator
03098851445



Composition



Inheritance



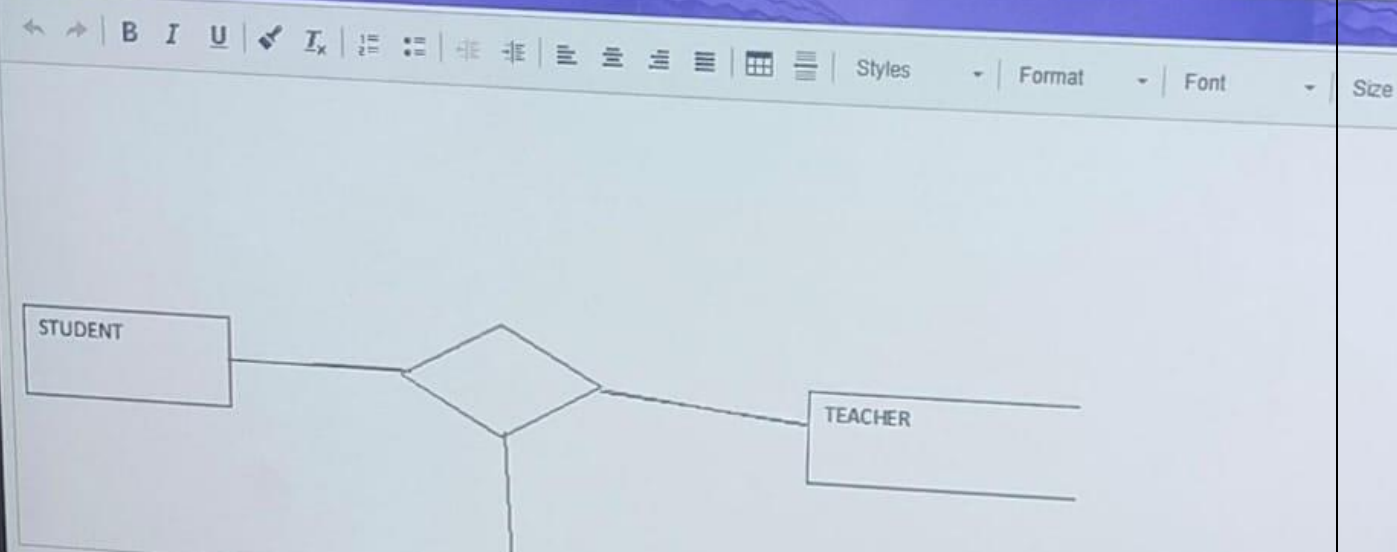
Specialization

CS304P (Object Oriented Programming (Practical))

Question No : 25 of 30

Suppose there are three classes, Student, Teacher, Course. Draw a UML class diagram to show a Ternary Association.

Answer



CS304P (Object Oriented Programming (Practical))

Question No : 24 of 30

A function template can be parameterized to operate on different types of _____.

Answer

☐ functions

☐ classes

☒ data

☐ objects

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 26 of 30

Write C++ code for a class named Point with two data members x and y of type int. Write a copy constructor for the Point class.

Answer

```
#include <iostream>

class point {
private:
    int x;
    int y;

public:
    // default constructor (good practice to include )
    point () : x ( 0 ) , y ( 0 ) {}

    // parameterized constructor
    point (int x_val, int y_val);

    x(x_val ), y (y_val) { }
```

CS304P (Object Oriented Programming (Practical))

Question No : 11 of 30

In case of private inheritance in C++, "public" data members of base class become ____ in derived class.

Answer

☒ private

Correct
Virtual Educator
03098851445

☐ protected

☐ public

☐ hidden

CS304P (Object Oriented Programming (Practical))

Question No : 27 of 30

Write C++ code to overload the pre-increment (++) operator for given class Point.

```
#include<iostream>
using namespace std;
class Point{
double x, y;
public:
```

// Write code to overload pre-increment operator here

Answer

```
#include <iostream>
using namespace std;

class point {
private:
    double x, y;
public:
    point (double x_val = 0, double y_val = 0) : x (x_val ), y(y_val) {} // constructor

    //overload the pre-increment
    operator (++)
    point& operator++( ) {
        ++x; // increment x
        ++y; // increment y
        return *this; // Return a reference to the modified object
    }
};
```

CS304P (Object Oriented Programming (Practical))

Question No : 28 of 30

Consider the following code segment:

```
class Complex{
private:
    double real, img;
public:
    // operator overloading code for subtraction Operator(-)
};
```

Answer

```
#include <iostream>
class complex {
private:
    double real , img ;

public:
    //constructor
    Complex (double r = 0.0 , double i = 0.0) : real (r), img (i) { }
    //overloading the subtraction operator (-)
    Complex operator -(const Complex& other) const {
        double result_real =real - other .real;

        double result_img =img _ other .img;
        return Complex(result_real, result_img);
    }
    //function to display complex number (for testing)
```

CS304P (Object Oriented Programming (Practical))

Question No : 10 of 30

Given below is the C++ code of the constructor of the Student class:

```
Student(int RNo)
{
    rollNo=RNo;
}
```

Identify the type of the constructor.

Answer

- ☐ Default Constructor
- ☒ Parametrized Constructor
- ☐ Deep Copy constructor
- ☐ Shallow Copy Constructor

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 29 of 30

12

Write a C++ code to:

- Create a class "Shape" and derive a class "Circle" from the "Shape" class.
- Shape class can have the function draw() but we cannot provide the implementation of function draw() in Shape.
- Every derived class must have implementation of draw(). You have to implement the draw() in Circle class that prints "Circle" on the screen.

Note: You cannot create the instance of the class Shape.

Answer

```
#include <iostream>

// abstract base class "shape"
class Shape {
public:
    // pure virtual function makes shape abstract
    virtual void draw ( )=0;
};

//derived class "circle"
class Circle : Public Shape {
```

Question No : 30 of 30

You are given a function `checkOddNumber(int num)` that throws an exception if the number is odd. You are required to write the missing try-catch block inside the `main()`

```
#include <iostream>
using namespace std;

// Function to check if a number is odd and throw an exception
void checkOddNumber(int num) {
    if (num % 2 != 0) {
        throw runtime_error("Exception: The number is odd!");
    }
}
```

Answer

```
#include <iostream>
#include <stdexcept> // include for runtime_error
using namespace std;

// Write the missing try-catch block here
try {
    checkOddNumber (number);
}
catch (const runtime_error& error ) {
    cerr << "Caught exception: " << error.what() << endl;
}
cout << "Program execution completed." << endl;
return 0;
}
```

CS304P (Object Oriented Programming (Practical))

Question No : 12 of 30

Which of the following type of iterator is used with Vector type container?

Answer

- ☐ Input Iterators
- ☐ Output Iterators
- ☐ Forward Iterators
- ☒ Random Access Iterators

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 7 of 30

In C++, while writing exception handling code, _____ handler must be preceded by a _____ block.

Answer

- ☐ if, else
- ☐ catch, throw
- ☐ catch, try
- ☒ try, catch

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 8 of 30

Which of the following is correct way to define a class template?

Answer

- ☐ `template< class T > void funName( ...);`
- ☐ `class XYZ { ... };template< typename T >`
- ☐ `template< Vector T > class XYZ { ... };`
- ☒ `template< class T > class XYZ { ... };`

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 9 of 30

Which of the following is correct way to create "vector" class instances for "char" data type?

Answer

☐ `< char > Vector charVector;`

☐ `Vector charVector< char >;`

☐ `Vector char charVector;`

☒ `Vector< char > charVector;`

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 4 of 30

Generic Algorithms made our code _____ of underlying data structure.

Answer

☐

specific

☒

independent

Correct
Virtual Educator
03098851445

☐

dependent

☐

constant

CS304P (Object Oriented Programming (Practical))

Question No : 5 of 30

Which of the following is the correct syntax to initialize the static data member 'x' (int type) of class Student?

Answer

☐ Student int::x= 0;

☐ Student::x= 0;

☒ int Student::x = 0;

☐ int Student.x=0;

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 6 of 30

Which of the following is the correct keyword to throw an exception in C++?

Answer

- ☐ throw_excep
- ☐ throw_exception
- ☒ throw
- ☐ exception

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 2 of 30

While working on Object Oriented Model, we can identify classes by extracting _____ from the problem statement.

Answer

- ☐ adjectives
- ☐ commands
- ☒ nouns
- ☐ verbs

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 1 of 30

Which of the following is not the type of STL Containers?

Answer

- ☐ Sequence Containers
- ☐ Associative Containers
- ☐ Container Adapters
- ☒ Iterator Containers

Correct
Virtual Educator
03098851445

CS304P (Object Oriented Programming (Practical))

Question No : 3 of 30



"lives in" is a(an) _____.

Answer

☐

Object

☐

Class

☒

Interaction

Correct
Virtual Educator
03098851445

☐

Entity

CS304P (Object Oriented Programming (Practical))

Question No : 28 of 30

```
class Complex{
private:
    double real, img;
public:
    // operator overloading code for subtraction Operator(-)
};
```

You are required to write C++ implementation of given overloaded subtraction operator.

Answer

```
double result_img = img - other_img;
return Complex(result_real, result_img);
}
//function to display complex number (for testing)
void display ( ) const{
    std :: cout << real << " + " << img << "i " << std :: endl;
}
};
int main () {
    Complex c1 (5.0, 3.0);
    Complex c2 (2.0 , 1.0);

    Complex c3 = c1 - c2 = " ;
    c3. display () ;
    return 0 ;
}
```

CS304P (Object Oriented Programming (Practical))

Question No : 28 of 30

Consider the following code segment:

```
class Complex{
private:
    double real, img;
public:
    // operator overloading code for subtraction Operator(-)
};
```

Answer

← → | **B** **I** U | T_x | | Styles

```
#include <iostream>
class complex {
private :
    double real , img ;

public:
    //constructor
    Complex (double r = 0.0 , double i = 0.0) : real (r), img (i) { }
    //overloading the subtraction operator (-)
    Complex operator -(const Complex& other) const {
        double result_real =real - other .real;

        double result_img =img _ other .img;
        return Complex(result_real, result_img);
    }
    //function to display complex number (for testing)
```

You are given a function `checkOddNumber(int num)` that throws an exception if the number is odd. You are required to write the missing try-catch block inside the `main()` function to handle the exception appropriately.

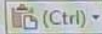
```
#include <iostream>
using namespace std;

// Function to check if a number is odd and throw an exception
void checkOddNumber(int num) {
    if (num % 2 != 0) {
        throw runtime_error("Exception: The number is odd!");
    } else {
        cout << "The number " << num << " is even." << endl;
    }
}

int main() {
    int number;
    cout << "Enter an integer: ";
    cin >> number;

    // Write the missing try-catch block here

    cout << "Program execution completed." << endl;
    return 0;
}
```



Consider the following code segment:

```
class Complex{  
private:  
    double real, img;  
public:  
    // operator overloading code for subtraction Operator(-)  
};
```

You are required to write C++ implementation of given overloaded subtraction operator.



I^{FE}

VIRTUAL