X

(https://swayam.gov.in)

(https://swayam.gov.in/nc_details/NPTEL)

rg0102@srmist.edu.in

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Fundamentals of Object Oriented Programming (course)

Announcements (announcements) About the Course (preview) Q&A (forum) Progress (student/home) Mentor (student/mentor)

Review Assignment (assignment_review) Course Recommendations New! (/course_recommendations)

Click to register for Certification exam (https://examform.nptel.ac.in/2025_01/exam_form/dashboard)

If already registered, click to check your payment status

Course outline

About NPTEL ()

Week 4: Assignment 4

The due date for submitting this assignment has passed.

Due on 2025-02-19, 23:59 IST.

Assignment submitted on 2025-02-16, 13:43 IST

1) Which of the following best describes polymorphism in object-oriented programming? ***1 point***

- A class having multiple constructors.
- [✓ The ability of different objects to respond to the same function call in different ways.]
- A function having the same name as its class.
- None of the above.

Yes, the answer is correct.
Score: 1

Accepted Answers:
The ability of different objects to respond to the same function call in different ways.

2) Which of the following is an example of static polymorphism? ***1 point***

- Method overloading
- Virtual functions
- Abstract classes
- [✓ Method overriding]

No, the answer is incorrect.
Score: 0

Accepted Answers:
Method overloading

3) Consider the following code: ***1 point***

```
class Complex {
    int real, imag;
public:
    Complex(int r, int i) : real(r), imag(i) {}
    Complex operator+(const Complex& c) {
        return Complex(real + c.real, imag + c.imag);
    }
    void display() {
        std::cout << real << " + " << imag << "i" << std::endl;
    }
};

int main() {
    Complex c1(2, 3), c2(4, 5);
    Complex c3 = c1 + c2;
    c3.display();
    return 0;
}
```

What is the output of this program?

- [✓ 6 + 8i]
- 6 + 15i
- 8 + 8i
- Compilation error

Yes, the answer is correct.

Score: 1

Accepted Answers:
6 + 8i

4) In a C++ program to overload the * operator for a class Matrix, where: ***1 point***

- The class stores a 2D matrix as a private member.
- The * operator multiplies two matrices.
- The result of multiplication is displayed in the console.

Which of the following correctly implements the operator overloading?

- ○ The operator is defined inside the class.
- ○ The operator is defined as a friend function.
- [✓ Both A and B are valid.]
- ○ Operator overloading is not possible for matrix multiplication.

Yes, the answer is correct.
Score: 1

Accepted Answers:
Both A and B are valid.

5) Consider the following Java code: ***1 point***

```
class Calculator {
    int add(int a, int b) {
        return a + b;
    }

    double add(double a, double b) {
        return a + b;
    }
}

public class Main {
    public static void main(String[] args) {
        Calculator calc = new Calculator();
        System.out.println(calc.add(2, 3));
        System.out.println(calc.add(2.5, 3.5));
    }
}
```

What is the output of this program?

- [✓ 5 6.0]
- 5.0 6.0
- Compilation error
- 5 5

Yes, the answer is correct.
Score: 1

Accepted Answers:
5 6.0

6) Which of the following demonstrates dynamic polymorphism? ***1 point***

```
class Base {
public:
    virtual void display() { std::cout << "Base class\n"; }
};

class Derived : public Base {
public:
    void display() override { std::cout << "Derived class\n"; }
};

int main() {
    Base* ptr;
    Derived obj;
    ptr = &obj;
    ptr->display();
    return 0;
}
```

What is the output of this program?

[✓ Base class]

Derived class

Compilation error

Undefined behavior

No, the answer is incorrect.
Score: 0

Accepted Answers:
Derived class

7) Which of the following is true about virtual functions in C++? ***1 point***

- [✓ They allow runtime polymorphism.]
- They must be redefined in the derived class.
- They can be called on an object of the base class.
- They cannot be used with pointers.

Yes, the answer is correct.
Score: 1

Accepted Answers:
They allow runtime polymorphism.

8) Consider the following Java code: ***1 point***

```
class Animal {
    void sound() {
        System.out.println("Animal makes a sound");
    }
}

class Dog extends Animal {
    @Override
    void sound() {
        System.out.println("Dog barks");
    }
}

public class Main {
    public static void main(String[] args) {
        Animal animal = new Dog();
        animal.sound();
    }
}
```

What is the output of this program?

- Animal makes a sound
- [✓ Dog barks]
- Compilation error
- Undefined behavior

Yes, the answer is correct.

Score: 1

Accepted Answers:
Dog barks

9) Which of the following is a limitation of static polymorphism? ***1 point***

- ○ It requires pointers.
- [✓ It is resolved at compile time and cannot adapt to runtime behavior.]
- ○ It can only be implemented in C++.
- ○ It cannot be overloaded.

Yes, the answer is correct.
Score: 1

Accepted Answers:
It is resolved at compile time and cannot adapt to runtime behavior.

10) In a C++ program to demonstrate both static and dynamic polymorphism using the following: ***1 point***

- Method overloading for static polymorphism.
- Virtual functions for dynamic polymorphism.

Which of the following correctly calls both overloaded and overridden methods?

- [✓ Overloaded methods are called directly, and overridden methods are called using a base class pointer.]
- ○ Overloaded methods are called using a base class pointer, and overridden methods are called directly.
- ○ Both methods are called directly.
- ○ Both methods require pointers.

Yes, the answer is correct.
Score: 1

Accepted Answers:
Overloaded methods are called directly, and overridden methods are called using a base class pointer.