National University of Computer & Emerging Sciences, Karachi
Spring-2024 School of Computing
Quiz 01
20th Feb 2024, 10:00 am – 11:00 am

Course No: CL1004	**Course Name: Object Oriented Programming**
Instructor Name/ Names:	
Student Roll No:	**Section:**

Time : 30 Min **Marks 10+15 points weightage 2.5**

Question 1: Observe and try to understand the following programs. Write errors if there are any available or write outputs if the programs are fine.

a.
```
class A
{
        public :
        int x=20;
};
class B
{
        public :
        int x=10;
};
int main()
{
        A obj1;
        B obj2;
        obj1 = obj2;
        cout<< obj1.x;
        cout<<endl;
return 0;
}
```

b.
```
class Point {
private:
int x, y;
public:
Point() : Point(1, 1) {}
Point(int i, int j) : x(i), y(j)
{
cout << x << " " << y << " - Normal
Constructor called" <<
endl;
}
Point(const Point& t) :
x(t.x), y(t.y) {
cout << y << " - Copy
Constructor called" <<
endl;
}
void setCoordinates(int i, int j) {
        x = i;
        y = j;
}
void print() const {
cout << "x: " << x << ", y: " << y << endl;
}
};
int main() {
        Point* t1, * t2;
        t1 = new Point(10, 15);
        t2 = new Point(*t1);
        Point t3 = *t1;
        Point t4;
        t4 = t3;
        t1->print();
        t2->print();
        t3.print();
        t4.print();
        Point t5 = *t1;
        t5.print();
        delete t1;
        delete t2;
return 0;
}
```

c.
```
class ConstInitializer {
public:
   ConstInitializer(int value) :constValue(value) {}
   void displayValue() const {
cout << "Const Value: " << constValue;
   }
private:
   const int constValue;
};
```

Question 02:

Evaluate your ability to design a Flight Booking System using C++ Flight Class:

Implement a C++ class named Flight to represent an individual flight holds some attributes flightNumber, departureAirport, destinationAirport, and int availableSeats.

Utilize techniques to keep the flight details and available seats no-access.

Booking Method: Create a public method within the Flight class: bool bookFlight(int numPassengers).

This method should handle the booking of a specified number of passengers, updating the available seats accordingly. Ensure that it returns true if the booking is successful and false if there are not enough available seats.

PassengerDevelop a Passenger class to represent an individual passenger.Include attributes passengerName and int passengerAge. Implement a public method within the Passenger class: void **bookTicket(Flight& flight, int numPassengers).**

This method should allow a passenger to book a specified number of tickets for a given flight, interacting with the Flight class.

create an instance of the Flight class, representing a specific flight.

Display the initial flight details, including the flight number, departure, destination, and available seats.

Create a passenger instance, allowing the passenger to book a certain number of tickets for the flight using the bookTicket method. Display the updated flight details after the booking.

Code Here: