# OOP (IT-2001)

# KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

## School of Computer Engineering



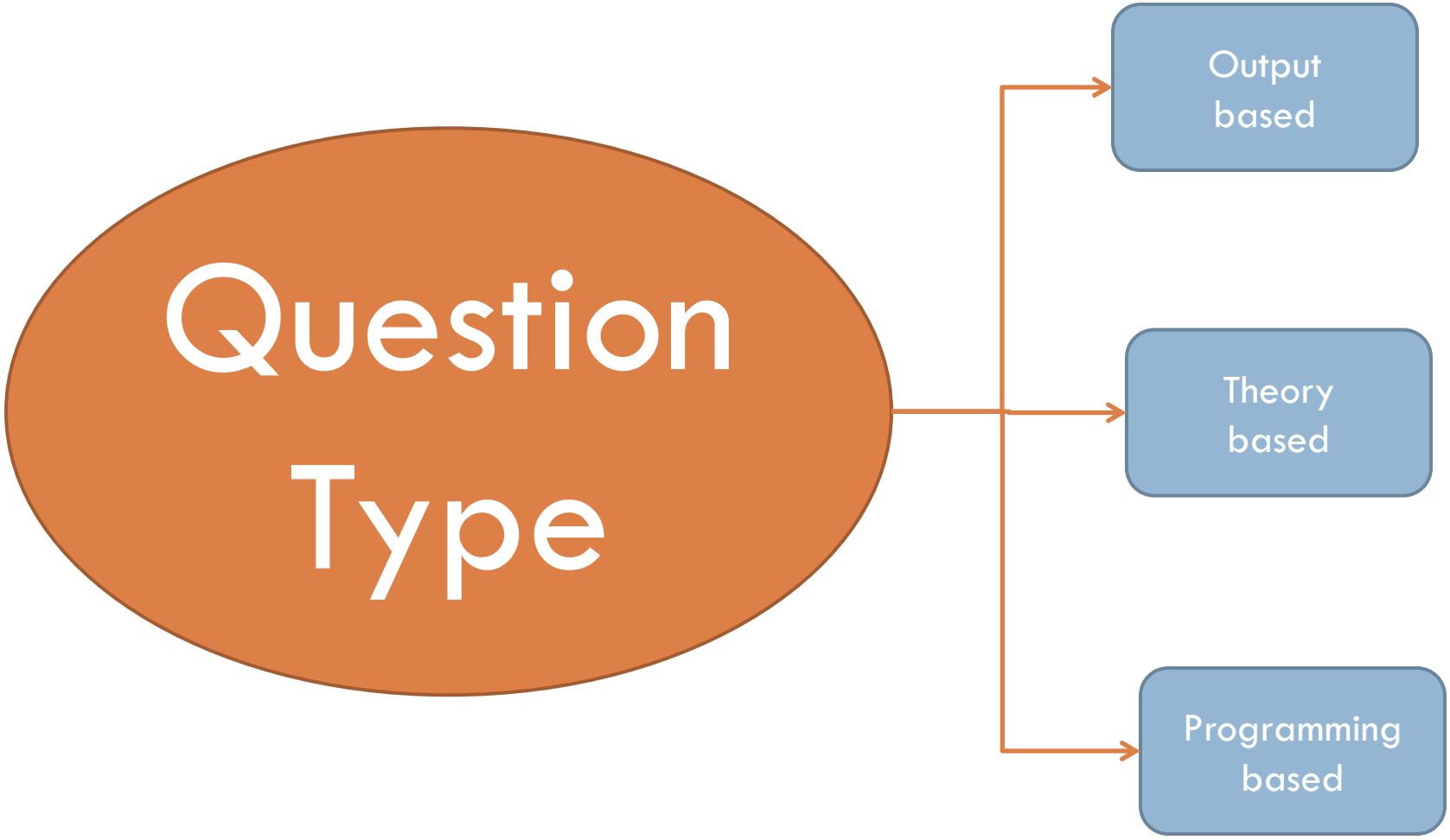

*3 Credit*

**Midterm Practice Questions**

**Mr. Rajat Behera - Associate Professor**

# Question Type





# Output Based Questions



```
#include<iostream>
using namespace std;
int x=2;
int main()
{
  int x=4;
  {
     int x=8;
     cout<< x;
     cout<<endl;
  }
  cout<< x;
  cout<<endl;
  cout<< ::x;
  cout<<endl;
 return 0;
}
```

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

int fun (int x, int y)
{
   return x + y ;
}
double fun (int x, int y)
{
   return x * y ;
}
int main()
{
   cout<<fun(5 , 10);
   return 0;
}
```

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

int main()
{
  float x=5.999;
  float *y,*z;
  y=&x;
  z=y;
 cout<<x<<","<<*(&x);
 cout<<","<<*y<<",";
 cout<<*z<<"\n";
 return 0;
}
```

# Output Based Questions cont…



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

int main()
{
        char *s="GoodLuck";
        for(int i=strlen(s)-1;i>=0;i--)
        {
                for(int j=0;j<=i;j++)
                cout<<s[j];
                cout<<endl;
        }
        return 0;
}
```

```
#include<iostream>
using namespace std;
class Test
{
   public:
   static int x;
   static void SetData(int xx) {
     x = xx;
   }
   static void Display() {
      cout<< x ;
   }
};
int Test::x = 0;
int main() {
   Test::SetData(44); Test::Display();
   return 0;
}
```

# Output Based Questions cont...



```
#include<iostream>
using namespace std;
class Test
{
   public:
   void GetData(char *s, int x, int y )
   {
      int i = 0;
      for (i = x-1; y>0; i++)
      {
         cout<< s[i];
         y--;
      }
   }
};
```

```
//continuation of  program
int main()
{
    Test testObj;
     testObj.GetData((char*)"Welcome!", 1, 3);
    return 0;
}
```

# Output Based Questions cont…



```
#include<iostream>
using namespace std;
class Test
{
    int x, y, z;
    public:
    Test(int xx, int yy, int zz)
    {
        x = ++xx;
        y = ++yy;
        z = ++zz;
    }
    void Show()
    {
        cout<< "" << x++ << " " << y++ << " " << z++;
    }
};
```

```
//continuation of  program
int main()
{
        Test objData(1, 2, 3);
        objData.Show();
        return 0;
}
```

# Output Based Questions cont…



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

class Test
{
   static int count;
   public:
   static void First(void)
   {
      count = 10;
   }
   static void Second(int x)
   {
      count = count + x;
   }
//continuation of  program
static void Display(void)
   {
      cout<< count << endl;
   }
};

int main()
{
      Test:: First();
      Test:: Second(5);
      Test :: Display();
       return 0;
 }
```

# Output Based Questions cont...



```
#include <iostream>
using namespace std;
class Sample
{
  int a;
  public:
  void set_data(int b){a=b;}
  void display(){cout<<a<<"\n";}
};
int main()
{
  Sample *s;
  s->set_data(10);
  s->display();
  return 0;
}
```

```
#include <iostream>
using namespace std;
class Display
{
    int cnt;
    public:
    Display(){cnt=0;}
    static void show_data(){cout<<++cnt<<"\n";}
};
int main()
{
   Display d;
   Display::show_data();
   return 0;
}
```

# Output Based Questions cont...



```
// Assume that integers take 4 bytes.
#include<iostream>
using namespace std;
class Test
{
  static int i;
  int j;
};

int main()
{
    cout << sizeof(Test);
    return 0;
}
```

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

int fun(int a, int b  = 1, int c =2)
{
    return (a + b + c);
}

int main()
{
    cout << fun(12, 2);
    cout << fun(12, , 2);
    return 0;
}
```

# Output Based Questions cont...



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

class Test
{
  private:
    int x;
  public:
    void setX (int x) { Test::x = x; }
    void print() { cout << "x = " << x << endl; }
};
```

```
//continuation of  program
int main()
{
    Test obj;
    int x = 40;
    obj.setX(x);
    obj.print();
    return 0;
}
```

# Output Based Questions cont...



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

class Test
{
    int value;
  public:
    Test(int v = 0) { value = v; }
    int getValue()  { return value; }
};

int main()
{
    const Test t;
    cout << t.getValue();
    return 0;
}
```

```
#include<iostream>
using namespace std;
class A
{
    int id;
  public:
    A (int i) { id = i; }
    void print () { cout << id << endl; }
};
int main()
{
    A a[2];
    a[0].print();
    a[1].print();
    return 0;
}
```

# Output Based Questions cont...



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

class A
{
   int id;
   static int count;
  public:
   A()
   {
      count++;
      id = count;
      cout << "constructor called " << id << endl;
   }
   ~A()
   {
      cout << "destructor called " << id << endl;
   }
};
```

```
//continuation of  program
int main()
{
   A a[2];
   return 0;
}
```

# Output Based Questions cont...



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

class A
{
   int aid;
public:
   A(int x)
   { aid = x; }
   void print()
   { cout << "A::aid = " <<aid; }
};

class B
{
    int bid;
public:
    static A a;
    B (int i) { bid = i; }
};

A B::a(10);

int main()
{
  B b(10);
  b.a.print();
  return 0;
}
```

# Output Based Questions cont...



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

int main()
{
  int i;
  for(i = 1; i <= 6; i += 2)
  {
    if(i%2 == 0)
     cout << i++ << "*\n";
    else
     cout << ++i << "#\n";
  }
  cout << i << "*\n";
  return 0;
}
```

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

void Position(int &c1, int c2 = 3)
{
   c1 += 2;
   c2 += 1;
}
int main()
{
   int p1 = 20, p2 = 4;
   Position(p1);
   cout << p1 << ", " << p2 << endl;
   Position(p2, p1);
   cout << p1 << ", " << p2 << endl;
}
```

# Output Based Questions cont...



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

void Withdef(int hisNum = 30)
{
   for (int i = 20; i <= hisNum; i += 5)
   cout << i << " ";
   cout << endl;
}

void Control(int & myNum)
{
   myNum += 10;
   Withdef(myNum);
}

int main()
{
   int yourNum = 20;
   Control(yourNum);
   Withdef();
   cout << "Number = " << yourNum;
}
```

# Output Based Questions cont…



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

struct POINT
{
   int X, Y, Z;
};
void StepIn(POINT & P, int Step = 1)
{
   P.X += Step;
   P.Y -= Step;
   P.Z += Step;
}
void StepOut(POINT &P, int Step = 1)
{
   P.X -= Step;
   P.Y += Step;
   P.Z -= Step;
}
```

```
int main()
{
   POINT P1 = {15, 25, 5}, P2 = {10, 30, 20};
   StepIn(P1);
   StepOut(P2, 4);
   cout << P1.X << ", " << P1.Y << ", " << P1.Z << "\n";
   cout << P2.X << ", " << P2.Y << ", " << P2.Z << "\n";
   StepIn(P2, 12);
   cout << P2.X << ", " << P2.Y << ", " << P2.Z << "\n";
}
```

# Output Based Questions cont...



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

int main()
{
   int Track[ ] = { 10, 20, 30, 40}, *Striker;
   Striker = Track;
   Track[1] += 30;
   cout << "Striker > " << *Striker << endl;
   *Striker -= 10;
   Striker++;
   cout << "Next@" << *Striker << endl;
   Striker += 2;
   cout << "Last@" << *Striker << endl;
   cout << "Reset To " << Track[0] << endl;
}
```

# Output Based Questions cont...



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

int main()
{
   int X[ ] = { 10, 25, 30, 55, 110 };
   int *p = X;
   while ( *p < 110)
   {
      if ( *p % 3 != 0)
         *p = *p + 1;
      else
         *p = *p + 2;
      p++;
   }
   //continuation of program
   for (int i = 4; i >= 1; i--)
   {
         cout << X[i] << "*";
         if (i % 3 == 0)
            cout << endl;
   }
   cout << X[0] * 3 << endl;
}
```

# Output Based Questions cont...



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

class METRO
{
   int Mno, TripNo, PassengerCount;
   public:
   METRO(int Tmno = 1)
   {
      Mno = Tmno;
      TripNo = 0;
      PassengerCount = 0;
   }

   void Trip(int PC = 20)
   {
      TripNo++;
      PassengerCount += PC;
   }

   void StatusShow()
    {
       cout << Mno << ":" << TripNo << ":" << PassengerCount;
     }
};

int main()
{
   METRO M(5), T;
   M.Trip();
   T.Trip(50);
   M.StatusShow();
   cout<<endl;
   M.Trip(30);
   T.StatusShow();
   cout<<endl;
   M.StatusShow();
}
```

# Theory Based Questions



1. Can you have a constructor with all default arguments? Justify your answer.
2. Differentiate between class and structure.
3. Why call by reference is preferred over call by value technique of passing object as an argument to a function?
4. What is a scope resolution operator? Discuss all the scenario where it can be useful.
5. When do we declare the data member of a class as static? Justify the need of static members along with an example.
6. Differentiate between friend function and friend class.
7. What do you mean by inline function? Discuss the advantages and disadvantages of using an inline function.
8. What is the use of a destructor and when it is going to be invoked?
9. How are data and functions organized in an object oriented system?
10. What do you mean by dynamic binding? How it is useful in OOP?
11. Differentiate between syntax errors and logical errors.
12. Differentiate between class and object.

# Theory Based Questions cont…



13. What are the header files? Why are they important? Can we write C++ program without using header file?
14. How can we get formatted output in C++ program?
15. Compare the use of if-else construct with ternary variable.
16. When will you prefer to use for, while, and do while loop?
17. Distinguish between break and continue statement.
18. Differentiate between function declaration and function definition.
19. Differentiate between formal parameters and actual parameters.
20. Explain the concept of recursive function with suitable examples.
21. Differentiate between an iterative function and a recursive function. Which one will be preferred to use in which circumstances?
22. How can we pass default arguments to a function? Explain with the help of an example.
23. Differentiate between call-by-value, call-by-reference and call-by-address.
24. How can a two-dimensional array be passed to the function.

# Theory Based Questions cont…



25. How an array of string is represented in the main memory with example?
26. Differentiate between array of character, a character pointer and a string.
27. How are strings read from the standard input device? Explain the different function used to perform string input operation.
28. Explain nested structure with suitable examples.
29. When union is used?
30. Differentiate between union and structure.
31. Is it possible to create an array of unions? Explain with the help of an example.
32. Is it possible to create an array of structures? Explain with the help of an example.
33. Is there any way to access the class private members without its object? Justify your answer
34. Define empty class.
35. Explain this pointer with suitable example.
36. Explain the techniques by which an object can be passed as an argument to a function.

# Theory Based Questions cont...



37. Explain the concept of dynamic memory allocation for array of objects with an example.
38. Explain friend class with suitable example.
39. Explain friend function with suitable example.
40. What is nested class? Explain its feature with suitable examples.
41. Explain constant member function with suitable function.
42. Explain static object with suitable example.
43. How can we return an object from a function?
44. How can a destructor called from a constructor. Illustrate with an example.
45. Is it mandatory to define a constructor for every classes? Justify with an example.
46. What is the significance of constructor and destructor.
47. Discuss the different types of constructors with suitable examples.
48. What are the techniques of invoking a copy constructor.
49. Why does the copy constructor accept the objects by reference and not by value
50. How to invoke the private constructor?

# Programming Based Questions

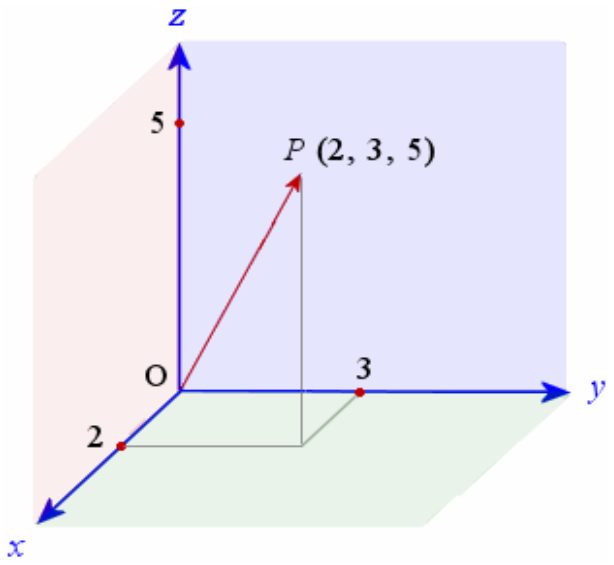


1. Write a menu driven program to read, display, simplify, add and subtract two rational numbers.
2. Write a menu driven program to read, display, add and subtract two complex numbers.
3. Write a menu driven program to read, display, add and subtract two distances.
4. Write a menu driven program to read, display, add and subtract two time objects.
5. Write a program that demonstrates the use of static data member, static member function and array of objects.
6. WAP that uses a friend function to swap numbers of a class.
7. WAP using Employee with members name, employee No, dept and salary. Demonstrate the use of inline functions and overloaded functions.
8. WAP to add, multiply two polynomial using classes and objects.
9. WAP that reads records of n students and sort them in descending order of their mark.

# Programming Based Questions



10. WAP to take the input of a faculty (ID, name, post, qualification, address, salary) and sort by name. Demonstrate the use of class and objects.
11. WAP to calculate simple and compound interest. Demonstrate the use of class and objects.
12. WAP to find the occurrences of the largest digit of a number within a set of numbers. Demonstrate the use of class and objects.
13. WAP using classes and objects to represent the vector in 3 D space and include the member functions to perform the following tasks:
    - ✓ Create the vector
    - ✓ Modify the value of a given axis
    - ✓ Multiply by scalar value for a given axis
    - ✓ Multiply by scalar value for all axes
    - ✓ Display the vector in the form P(10, 20, 30)
    - ✓ Divide by scalar value for all axes
    - ✓ Add by scalar value for all axes



# Programming Based Questions



14. WAP using classes, objects and constructors to represent the class student with following specification

- Private members
  - ✓ Roll no ✓ Name ✓ 6 subjects mark ✓ Total
- Public members function to
  - ✓ Calculate the total marks
  - ✓ Display the grade
  - ✓ Accept values from input device to calculate total and display grade.

15. WAP using classes, objects and constructors to represent the bank account with the following specification

- Private members
  - ✓ Name of the depositor ✓ Type of account
  - ✓ Account number ✓ Balance amount in the account
- Public members function to
  - ✓ Assign initial values ✓ Withdraw an amount after checking balance
  - ✓ Deposit an amount ✓ Display name and balance

# Programming Based Questions



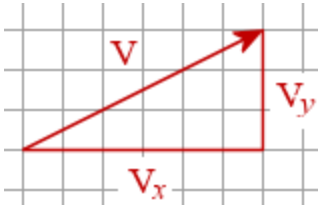
16. Define a class called Point that stores the x and y coordinates of the point. WAP that uses parameterized constructor for initializing the class objects and also display the coordinates.
17. Define a class called Complex that stores real and imaginary part of the complex number. WAP that uses overloaded constructors for initializing the class objects and also display the part.
18. WAP using appropriate constructors and destructors to represent the faculties working in an organization with the following specification

- Private members
  - ✓ Name
  - ✓ Employee ID
  - ✓ Salary
  - ✓ Designation
  - ✓ Date of Birth
  - ✓ Address
- Public members function to
  - ✓ Assign initial values
  - ✓ Deposit salary
  - ✓ Withdraw an amount after checking balance
  - ✓ Display name, Designation and Age

# Programming Based Questions



19. WAP using appropriate constructors and destructors to represent the vector in 2 D space and include the member functions to form the following tasks:
    - Create the vector
    - Modify the value of a given element
    - Multiply by scalar value
    - Display the vector in the form (10, 20)




20. A book shop maintains the inventory of books that are being sold at the workshop. The list includes details such as author, title, price, publisher and stock position. Whenever a customer wants a book, the sales person inputs the title and author and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed. If it is, then the system displays the book details and requests for the number of copies required. If the requested copies are available, the total cost of the requested copies is displayed otherwise the message "Required copies not in stock" is displayed. WAP using a class called Books with suitable member functions and constructors.

# Programming Questions



21. WAP to dynamically allocate memory to string objects. Use copy constructor to copy one string into another.
22. WAP to keep a track of number of objects created, number of objects destroyed and number of active objects in a program.
23. WAP that uses a date structure within a class. Enter any date and your birth date. The program must display your exact age in years, months and days.
24. WAP that uses a class within a class.
25. WAP that reads records of n students and find the
    - ✓ average mark of each student
    - ✓ # of students above average mark in the class.
    - ✓ # of students below average mark in the class.
    - ✓ Sort students in ascending order of their mark.
    - ✓ Display the name of the student secured highest mark.
    - ✓ Display the roll number of the student secured highest mark from bottom.


SUCCESS
All the best for
your Exams