

Registration No. :

--	--	--	--	--	--	--	--	--	--

Total number of printed pages – 16

BCSE 3201/BECS 2207/PCCS 2207/BECS 2212 (N/O) B. Tech

Third Semester Examination – 2010

OBJECT ORIENTED PROGRAMMING (CSE/IT)/C++ AND OBJECT ORIENTED PROGRAMMING
(New and Old Course)

Full Marks – 70

Time : 3 Hours

(Students are required to give their answer any one Course
according to the Syllabus)

OBJECT ORIENTED PROGRAMMING (CSE/IT)
(NEW COURSE)

Answer Question No. 1 which is compulsory and any five from the rest.

The figures in the right-hand margin indicate the marks.

1. Answer the following questions : 2 × 10

(a) What will be the output and why ?

```
#include<iostream>
using namespace std;
int main()
{ const int a=20;
  const int *ptr=&a;
  cout<<*ptr<<endl;
  (*ptr)++;
  cout<<a<<endl;
  return 0;
}
```

(b) State at least two differences between an inline function and a macro substitution.

(1)

P.T.O.

- (c) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{ int i;
public:
    static void f1( )
        { i=20; i++; }
    void f2( )
        { cout<<"i"<<endl; }
};
int main( )
{
    X ob1,ob2;
    ob1.f1( );
    ob2.f2( );
    return 0;
}
```



- (d) Explain the two primary differences between a pointer variable and a reference variable.
- (e) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{
public:
    int p;
};
class Y:protected X
{
public:
    int q;
};
int main()
{
    Y yob;
    yob.p=10;
    yob.q=20;
    cout<<yob.p<<" "<<yob.q<<endl;
    return 0;
}
```

(f) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{
public:
void f1()
{cout<<"X-Men"<<endl;}
virtual void f2()=0;
};
class Y:public X
{
public:
void f1()
{cout<<"Y-Men"<<endl;}
};
int main()
{
X *xp;
Y yob;
xp=&yob;
xp->f1();
return 0;
}
```

(g) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{
int i;
public:
static void f(int i){ this->i=i; }
};
int main( )
{
X xob;
xob.f(20);
return 0;
}
```

(h) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{
    const int i;
    public:
        X(int p){i=p;}
        void f(){cout<<"i="<<i<<endl;};
}
int main( )
{
    X xob(100);
    xob.f();
    return 0;
}
```



(i) What will be the output and why ?

```
#include<iostream>
using namespace std;
int main()
{
    cout<<"Inside main"<<endl;
    try
    {
        cout<<"Inside try block"<<endl;
        throw 22;
        cout<<"Exception thrown"<<endl;
    }
    catch(double d)
    {
        cout<<"Caught an exception="<<d;
    }
    cout<<"End"<<endl;
    return 0;
}
```

- (j) What will be the output and why ?

```
#include<iostream>
using namespace std;
template<class T1> void f(T1 tt1)
{ cout<<"tt1="<<tt1<<endl;}
template<class T2> void f(T2 tt2)
{ cout<<"tt2="<<tt2<<endl;}
int main()
{
f(1000);
  f(101.7);
  return 0;
}
```



2. (a) Create a class called **student** which contains protected attributes such as stud_name, stud_roll and stud_branch. Provide an appropriate method to take user input to initialize these attributes and display the details regarding 50 students of a class. 5
- (b) Create a class called **Area** which contains a method called "find_area". Write down appropriate code to create objects named as **circle** and **rectangle** of the above class and implement **function overloading** to calculate area of a rectangle and area of a circle based upon user input. 5
3. (a) Write a C++ program to convert a primitive or built in type value such as int or char to a user defined type value such as class A or class X. 5
- (b) Write a C++ program where you can only be able to create a single instance of a class. Upon trying to create more than one number of instances will result in termination of the program. 5
4. (a) Create a class called **Point** with two integer attributes such as x and y to represent its x-coordinate and y-coordinate. Provide constructor to initialize the attributes. Provide another method named as move() which will move the coordinates only in the direction of x-axis for 10 units incrementing 1 unit at a time. Also display the new and old values of the coordinates. 5
- (b) With an appropriate example, explain the role of virtual base class in removing ambiguities in case of diamond inheritance which is a special case of multi path inheritance. 5

5. (a) Create an abstract class called **Figure** which contains a pure virtual function called `find_area()` and a protected attribute named as `area`. Create two new derived classes from the above class named as **Circle** and **Square** having double type attribute named as `radius` and `side` respectively. Implement **dynamic polymorphism** to find out area of a circle and a square, and show the result. 5
- (b) Write appropriate code to overload the pre increment and post increment operators in a same program using non member operator functions. 5
6. (a) Suppose there is a class called **X** with a double type attribute. Write a C++ program to create three objects named as `ob1`, `ob2` and `ob3` of the above class and perform the following operations. 5
- `ob2 = 5.5 + ob1;`
`ob3 = ob1 + 6.7;`
- (b) Write a complete program to create a class named as **Student** with protected attributes such as `id` and `marks`. The attributes should be initialized only through constructors. The class contains a public method named as `show()` to display the initialized attributes. Provide a mechanism to create an array of **Student** objects. The array size should be given by the user at run time. 5
7. (a) Provide at least one good reason to choose a **const reference** rather than a **reference** variable as a parameter to a **copy constructor**. Explain with a suitable example. 5
- (b) Write a complete program where you can achieve the following conversion within `main()`: 5
- `A ob1(10), ob2(101.5);`
`int i=ob1;`
`double d=ob2;`
where **A** is a user defined class.
8. (a) Write a complete program where you can restrict a user defined function to throw only `int` or `char` type exception out of it. 5
- (b) Write a complete program to declare and define a generic function that is capable of arranging any kind of elements in descending order. 5



**C++ AND OBJECT ORIENTED PROGRAMMING
(NEW COURSE)**

Answer Question No. 1 which is compulsory and any five from the rest.

The figures in the right-hand margin indicate marks.

1. Answer the following questions :

2 × 10

(a) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{
    int p;
    public:
        X(int a){p=a;}
        void f(){cout<<"p="<<p<<endl;}
};

int main()
{
    X xob;
    xob.show();
    return 0;
}
```



(b) State the differences between the following two declarations with example :

const int *p; and int *const p;

(c) What will be the output and why ?

```
#include<iostream>
using namespace std;
class A
{ int i;
    public:
        static void f( ) { i=20; i++; }
};

int main( )
{ A ob1,ob2;
    ob1.f( );
    ob2.f( );
    return 0;
}
```

(d) private attributes can't be inherited. State a remedy for this problem so that attributes of a class behave like private attributes but can be inherited. Explain with an example.

(e) What will be the output and why ?

using namespace std;

```
struct X
```

```
{int p;};
```

```
struct Y:protected X
```

```
{int q;};
```

```
int main()
```

```
{
```

```
    Y yob;
```

```
    yob.p=11;
```

```
    yob.q=22;
```

```
    cout<<yob.p<<" "<<yob.q<<endl;
```

```
    return 0;
```

```
}
```



(f) What will be the output and why ?

using namespace std;

```
class X
```

```
{
```

```
    int p;
```

```
    public:
```

```
        friend void f(int a);
```

```
};
```

```
void f(int a)
```

```
{
```

```
    p=a;
```

```
    cout<<"p="<<p<<endl;
```

```
}
```

```
int main()
```

```
{
```

```
    f(200);
```

```
    return 0;
```

```
}
```


(g) What will be the output and why ?

```
using namespace std;
class X
{
    int a;
    public:
        X(int p){a=p;}
};
class Y
{
    public:
        int b;
};
class D:public X, public Y
{
    public:
        D(int k){b=k;}
        void show(){cout<<"b="<<b<<endl;}
};
int main()
{
    D dob(100);
    dob.show();
    return 0;
}
```



(h) State at least two differences between new and malloc.

(i) What will be the output and why ?

```
using namespace std;
class X
{
    int i;
    public:
        X(int a){i=a;}
        X(X &copy){i=copy.i;}
        void f() { cout<<"i="<<i<<endl; }
};
int main( )
{
    X const xob1(10);
    X xob2=xob1;
    xob2.f();
    return 0;
}
```

(j) What will be the output and why ?

```
using namespace std;
void f(int a) throw()
{
    if (a==1) throw 100;
    if (a==2) throw 20.5;
}
int main()
{
    cout<<"Inside main"<<endl;
    try
    {
        cout<<"Inside try block"<<endl;
        f(2);
    }
    catch(int i) {cout<<"i="<<i<<endl;}
    catch(double d){cout<<"d="<<d;}
    return 0;
}
```



2. (a) Create a class called **Employee** which contains protected attributes such as emp_id, emp_salaray and emp_da. emp_da is 20% of the emp_salary. Provide an appropriate method to take user input to initialize the attributes and display the details regarding 25 students of a class. 5
- (b) Create a class called **Volume** which contains a method called "find_vol". Write down appropriate code to create objects named as **sphere** and **cylinder** of the above class and implement **function overloading** to calculate volume of a sphere and cylinder based upon user input. 5
3. (a) Write a C++ program to convert a primitive or built in type value such as int or char to a user defined type value such as class A or class X. 5
- (b) Write a C++ program where you can only be able to create a single instance of a class. Upon trying to create more than one number of instances will result in termination of the program. 5

4. (a) With an appropriate example, explain how ambiguities can be resolved for public and protected attributes in case of multi path inheritance without using virtual base class. 5

(b) Create an abstract class called **Shape** which contains a pure virtual function called `find_vol()` and a protected attribute named as `volume`. Create two new derived classes from the above class named as **Cube** and **Sphere** having double type attribute named as `side` and `radius` respectively. Implement **dynamic polymorphism** to find out volume of a cube and a sphere. Also display the result. 5

5. (a) Write appropriate code to overload the bit-wise `<<` operator to perform the following task :

```
int y = x<<ob;
```



where `ob` is an object of class `myclass` with a single `int` attribute and `x` is an `int` declared in `main()` and its value is to be supplied by the user. 5

(b) Suppose there is a class called **X** with two double type attributes. Write a C++ program to create two objects named as `ob1` and `ob2` of the above class and overload the binary `==` operator to perform the following operation within `main()` : 5

```
if(ob1== ob2) cout<<"Objects are same"<<endl;
```

```
else cout<<"Objects are different"<<endl;
```

6. (a) Suppose there is a class called **X** with a double type attribute. Write a C++ program to create three objects named as `ob1`, `ob2` and `ob3` of the above class and perform the following operation : 5

```
ob2 = 5.5 + ob1;
```

```
ob3 = ob1 + 6.7;
```

- (b) Write an appropriate C++ code showing ambiguity resolving mechanism where a class attribute has same name as that of a local parameter of a member function by using this pointer. 5
7. (a) Write a complete program to create a class called **Account** with protected attributes such as account number and balance. The attributes should be initialized through constructors. The class contains a public method named as show() to display the initialized attributes. Provide a mechanism to create an array of **Account** objects. The array size should be given by the user at run time. 5
- (b) What is a **copy constructor** ? Explain the role of a **copy constructor** while initializing a pointer attribute of a class for which the memory allocation takes place at the run time. 5
8. (a) Write a complete program where you can perform the following conversion within main() : 5

A ob1(5.5), ob2;

int i=10;

double d=ob1;

ob2= i; where **A** is a user defined class.

- (b) Write a complete program where an **exception** of type double can be **rethrown**. 5

**C++ AND OBJECT ORIENTED PROGRAMMING
(OLD COURSE)**

*Answer Question No. 1 which is compulsory and any **five** from the rest.*

The figures in the right-hand margin indicate marks.

1. Answer the following questions :

2 × 10

(a) What will be the output and why ?

```
#include<iostream>
using namespace std;
class X
{
    int p;
    public:
        X(int a){p=a;}
        void f(){cout<<"p="<<p<<endl;}
};

int main()
{
    X xob;
    xob.show();
    return 0;
}
```



- (b) State the differences between the following two declarations with example :
const int *p; and int *const p;
- (c) Describe the restrictions applied to static methods of a class. Give suitable example.
- (d) Private attributes can't be inherited. State a remedy for this problem so that attributes of a class behave like private attributes but can be inherited. Explain with an example.
- (e) State at least two differences between an inline function and a macro substitution.

(f) What will be the output and why ?

```
using namespace std;
class X
{
    int p;
public:
    friend void f(int a);
};
void f(int a)
{
    p=a;
    cout<<"p="<<p<<endl;
}
int main()
{
    f(200);
    return 0;
}
```



- (g) Suppose a class contains a reference variable of type int. Explain how that reference variable can be initialized.
- (h) State at least two differences between new and malloc.
- (i) What is an explicit constructor ? Give an example.
- (j) What is a generic catch block ? Explain with an example.
2. (a) Create a class called Triangle with 3 sides as its protected attributes. Initialize the attributes with help of a constructor. Provide a method to calculate the area of a Triangle object and display the result. 5
- (b) Create a class called Employee with protected attributes such as emp_name, emp_id and emp_basic_pay. Initialize the attributes with the help of a constructor. Provide a method to calculate the gross salary of 5 employees as per the details given below and also display the details : 5
- dearness allowance (DA) = 50% the basic pay,
house rent allowance (HRA) = 20% of the basic pay,
conveyance allowance (CA) = 5% of the basic pay,
professional tax (PT) = Rs. 200.00 and income tax (IT) = 10% of net pay
where,
net pay = basic pay + DA + HRA. Gross pay = basic pay + DA + HRA + CA
– PT – IT.

3. (a) Write a complete C++ program to create a singleton class. A singleton class is a class for which only one instance can be created. 5
- (b) Write a C++ program where you can count the number of instances created for a class. 5
4. (a) With an appropriate example, explain how ambiguities can be resolved for public and protected attributes in case of multi path inheritance without using virtual base class. 5
- (b) Create an abstract class called Shape which contains a pure virtual function called find_vol() and a protected attribute named as volume. Create two new derived classes from the above class named as Cube and Sphere having double type attribute named as side and radius respectively. Implement dynamic polymorphism to find out volume of a cube and a sphere. Also display the result. 5
5. (a) Write a complete C++ program to overload the pre increment operator using a non member operator function. 5
- (b) Suppose there is a class called X with two double type attributes. Write a C++ program to create two objects named as ob1 and ob2 of the above class and overload the binary == operator to perform the following operation within main() :
- if(ob1 == ob2) cout<<"Objects are same"<<endl;
 else cout<<"Objects are different"<<endl;
6. (a) With the help of a suitable example explain the role of a virtual destructor. 5
- (b) Write an appropriate C++ code showing ambiguity resolving mechanism where a class attribute has same name as that of a local parameter of a member function by using this pointer. 5
7. (a) Write a complete program to create a class called Account with protected attributes such as account number and balance. The attributes should be initialized through constructors. The class contains a public method named as show() to display the initialized attributes. Provide a mechanism to create an array of Account objects. The array size should be given by the user at run time. 5



- (b) With a suitable example explain deep copying of objects. What is the role of a copy constructor in deep copying? 5
8. (a) Write a C++ program to overload a generic function along with specific version of the same function. 5
- (b) Write a complete program where an exception of type char can be rethrown. 5

