

**2.OBJECT ORIENTED PROGRAMMING
&
3. FUNCTION OVERLOADING**

2010 Delhi:

2. (a) What do you understand by Polymorphism? Also, give an example in C++ to illustrate the same. **2**

Ans. The process of using an operator or a function in different ways for different set of inputs given is known as polymorphism. Function overloading is an example of polymorphism, where the functions having same name with different set of parameters perform different operations.

Example:

```
void Disp ( ) //Function 1
{
    cout<<"Hello"<<endl;
}
void Disp(int N) //Function 2
{
    for(int I=1;I<=N;I++)
        cout<<I<<endl;
}
void Disp (int N,int M) //Function 3
{
    for (int I=N;I<=M; I++)
        cout<<N<<"x"<<I<<"="<<N*I<<endl;
}
void main ( )
{
    int x=5, y=10;
    Disp(x); //Function 2 called-Prints numbers from 1 to 5
    Disp(x,y) ; //Function 3 called- Prints from multiples of 5
                //ranging from 5 to 10
    Disp ( ) ; //Function 1 called- Prints Hello
}
```

2010 Outside Delhi:

2. (a) What do you understand by Data Encapsulation and Data Hiding ? Also, give an example in C++ to illustrate both. **2**

Ans. Data Encapsulation: Wrapping up of data and functions together in a single unit is known as Data Encapsulation. In a class, we wrap up the data and functions together in a single unit.

Data Hiding: Keeping the data in private/protected visibility mode of the class to prevent it from accidental change is known as Data Hiding.

```
class Computer
{
    char CPU[10] ;int RNM; //Data Hiding
public:
    //Data
    Encapsulation
    void STOCK();
    void SHOW();
};
```

2009 Outside Delhi:

2. (a) What is function overloading? Give an example in C++ to illustrate function overloading. **2**

Ans Function overloading is an example of polymorphism, where the functions having same name with different set of parameters perform different operations.

OR

When a function is having more than one definition and differentiable by the number/type of parameters is known as function overloading

Example:

```
void Disp() //Function 1
{
    cout<<"Hello"<<endl;
}
void Disp(int N) // Function 2
{
    for (int I=1;I<=N;I++)
        cout<<I<<endl;
}
void main ( )
{
    int x=5;
    Disp(x); //call for Function 2 - Prints
            //numbers from 1 to 5
    Disp(); //call for Function 1 - Prints Hello
}
```

2005 Delhi:

2.a) Define the term Data Hiding in the context of Object Oriented Programming. Give a suitable example using a C++code to illustrate the same. **2**

Ans: A class groups its members into three sections: private, protected and public. The private and protected members remain hidden from outside world. Thus through private and protected members, a class enforces data – hiding. (The outside world is given only the essential and necessary information through public members, rest of the things remain hidden, which is nothing but abstraction. The act of representing only essential features without including background details is known as abstraction.)

Eg: class ABC

```
{
    private: int a,b;
    protected: int c,d;
    public: int e,f;
            void disp( )
            {
                ----
                ----
            }
            ----
}
```

In the above class public members (ie e,f and disp()) only will be available to outside the class.. The other private members (a,b), protected members (c,d) will not be available to

outside the class. This concept is called data hiding.

2005 Outside Delhi:

2.a) Define the term Data Encapsulation in the context of Object Oriented Programming. Give a suitable example using a C++ code to illustrate the same.

2

Ans: Encapsulation is wrapping up of characteristics and behavior into one unit. While implementing encapsulation, following things are taken care:

- a) Anything that an object does not know or cannot do is excluded from the objects.
- b) Encapsulation is used to hide unimportant implementation details from other objects.
- c) Packaging an object's variables within the protective custody of its methods is called encapsulation and this task is accomplished through classes. Ie the data and associated functions are wrapped up in one unit called class.

A class binds together data and its associated functions under one unit thereby enforcing encapsulation.

Eg: class Rectangle

```
{ private: float len,bre,area;
public: void readData( )
{ cout<<"\nEnter the length and
breadth..";
cin>>len>>bre;
}
void calculate( )
{ area=len*bre;
}
void display( )
{ cout<<"\nThe area of the rectangle =
"<<area;
}
};
```

Eg: Here in the above class the data members ie len,bre,area and the member functions ie readData(), calculate(), display() are bind together in a class named as Rectangle. Ie The member functions can access any data member in the class.

Benefits with encapsulation:

- (i) Modularity.
- (ii) Information hiding.

2004:

1.a) What is polymorphism? Give an example in C++ to show its implementation in C++.

Ans: Polymorphism is the attribute that allows one interface to be used with different situation. C++ implements polymorphism

through virtual functions, through overloaded functions and overloaded operators.

A virtual function is used to specify the interface in abstract class, but its implementation details are made available by the concrete class(es).

An overloaded function refers to a function having (one name and) more than one distinct meanings. Similarly, when two or more distinct meanings are defined for an operator, it is said to be an 'overloaded operator'. It is the compiler's job to select the specific action as it applies to each situation.

Eg: The program in the next answer.

2003:

2.a) What do you understand by function overloading? Give an example illustrating its use in a c++ program.

Ans: A function name having several definitions that are differentiable by the number or types of their arguments, is known as an overloaded function and this process is known as function overloading.

Function overloading not only implements polymorphism but also reduces number of comparisons in a program and thereby makes the program run faster.

Example program illustrating function overloading:

//Program to find out area of a circle or area of //rectangle using function overloading.

```
#include<iostream.h>
#include<conio.h>
void area(float r)
{ cout<<"\nThe area of the circle = "<<3.1415*r*r;
}
void area(float l,float b)
{ cout<<"\nThe area of the rectangle = "<<l*b;
}
void main( )
{ float rad,len,bre;
int n;
clrscr();
cout<<"\n1. Area of a Circle...";
cout<<"\n2. Area of a Rectangle...";
cout<<"\n\nEnter your choice: ";
cin>>n;
switch(n)
{case 1: cout<<"\nEnter the radius: ";
cin>>rad;
area(rad);
break;
```

```

case 2: cout<<"\nEnter the length
        and breadth: ";
        cin>>len>>bre;
        area(len,bre);
        break;
default: cout<<"\nYou have to enter
        either 1 or 2";
} //end of switch
getch();
}

```

2000:

1.a) Illustrate the concept of function overloading with the help of an example. 1

Ans: The above answer.

1998:

1.a) Define the following terms:

(i) Inheritance (ii) Encapsulation.

Ans:a) Inheritance: The capability of one class to inherit properties from another class is called as inheritance. The class inheritance, lets you generate a model that is closer to the real world. The class inheritance lets you derive new classes (derived class) from old ones (base class), with the derived class inheriting the properties, including the methods of the old class.

Uses of Inheritance:

i) Capability to express the inheritance relationship which ensures the closeness with the real world models.

ii) Reusability.

iii) Transitive nature of inheritance.

b) Encapsulation: The wrapping up of data and functions into a single unit (class) is called as encapsulation.

Model Paper 1 for 2008-09 Batch:

Q2.(a) What do you understand by Data Encapsulation and Data Hiding? 2

Answer:

Data Encapsulation: Wrapping up of data and function together in a single unit is known as Data Encapsulation. In a class, we wrap up the data and function together in a single unit.
 Data Hiding: Keeping the data in private visibility mode of the class to prevent it from accidental change is known as Data Hiding.

```

class Computer
{

```

```

char CPU[10];
int RAM;
public:
void STOCK();
void SHOW();
};

```

Data Encapsulation

Model Paper 2 for 2008-09 Batch:

1. (a) What is the difference between Object Oriented Programming and Procedural Programming? 2

Answer:

| Object Oriented Programming | Procedural Programming |
|---|--|
| <ul style="list-style-type: none"> • Emphasis on Data • Follows Bottom-Up approach in program design • Data hiding feature prevents accidental change in data • Features like data encapsulation, polymorphism, inheritance are present | <ul style="list-style-type: none"> • Emphasis on doing things (functions) • Follows Top-down approach in program design • Presence of Global variables increase chances of accidental change in data • Such features are not available |

2.(a) What do you understand by Polymorphism? Give a suitable example of the same. 2

Answer:

Polymorphism: It is a method of using the same operator or function (method) to work using different sets of input. Function overloading is one of the example of polymorphism, where more than one function carrying same name behave differently with different set of parameters passed to them.

```

void Display()
{
    cout<<"Hello!"<<endl;
}
void Display(int N)
{
    cout<<2*N+5<<endl;
}

```

Sample Paper 1 for 2009-10 Batch:

2.a) What do you understand by Data Encapsulation and Data Hiding? Also, give a suitable C++ code to illustrate both. 2

A) Data Encapsulation: Wrapping up of data and functions together in a single unit is known as Data Encapsulation. In a class, we wrap up the data and functions together in a single unit.

**“The moment which is lost,
is lost for ever.
So utilize the time
properly”**

Data Hiding: Keeping the data in private visibility mode of the class to prevent it from accidental change is known as Data Hiding.

```
class Computer
{
    char CPU[10];int RAM; //Data Hiding
public:
    //Data
    Encapsulation
    void STOCK();
    void SHOW();
};
```

Sample Paper for 2009-10 Batch:

2.a) What do you understand by Polymorphism?
Give a suitable example of the same. 2

Ans) Polymorphism: It is a method of using the same operator or function (method) to work using different set of inputs. Function overloading is one of the examples of polymorphism, where more than one function carrying same name behave differently with different set of parameters passed to them.

```
void Display()
{
    cout<<"Hello!"<<endl;
}
void Display(int N)
{
    cout<<2*N+5<<endl;
}
```

TEST

1. What do you understand by Polymorphism? Also, give an example in C++ to illustrate the same.
2. What is function overloading? Give an example in C++ to illustrate function overloading.
3. Define the term Data Hiding in the context of Object Oriented Programming. Give a suitable example using a C++ code to illustrate the same
4. What do you understand by Data Encapsulation and Data Hiding ?' Also, give an example in C++ to illustrate both.