

Assignment - 20

Q1] Draw object layout of below code and explain the internal working in detail

1]

class base

{

public:

int i;

float f;

double d;

void fun();

void gun();

};

class Derived : public base

{

public:

int i;

double d;

void sun();

void fun();

};

int main()

{

base bobj;

Derived dobj;

return 0;

};

- In the above code single inheritance is implemented in that class base, having the characteristics i, f and d & behaviours func1 & func2.
- Class derived is inherit the base class means class derived acquire all the properties of base class.
- In main fu-method we create the object of base class as well as Derived class.
- Using the object of base class we access only properties of base class but by using the hello class object we access all properties of both class.

Object Layout

bobj

100	base::i	100	base::i
104	base::f	104	base::f
108	base::d	108	base::d
116		116	Derived::i
		120	Derived::f
		128	Derived::d

2) class base1

```
public:
    int i;
    float f;
```

```
void fun()
```

class base2

```
public:
    int j;
    float g;
    void func();
```

class Derived : Public base1, base2

Public:

```
int i;
double d;
```

```
void sun()
```

```
void fun()
```

int main()

```
Derived dobj;
```

```
return 0;
```

}

In the above inheritance Application multiple class base1 having i, f & behaviours class base2 having j, g and behaviour func. class Derived inherit the class base and base2 having i and d behaviours. In main method we create the object of Derived class i.e dobj By using this object we access the all properties of all classes.

object layout

100		base1:: i
104		base1:: f
108		base2:: j
112		base2:: g
116		Derived:: i
120		Derived:: d
128		

3) class base
 int i;
 float f;
 void func() { 1000 }
 virtual void gun() { 2000 }

class Derived : public base
public :
 int i;
 double d;
 virtual void func() { 3000 }
 virtual void gun() { 4000 }
 virtual void sun() { 5000 }

int main()
{
 Derived dobj;
 return 0;
}

M	T	W	T	F	S	S
Page No:						
Date:						

YOUVA

In the above Application base class contain characteristics i.e. f & g behaviours func & also virtual function i.e. gunc
 class Derived inherit the base class having characteristics if d behaviour virtual behaviours func, sun() & the normal behaviours gunc
 In main method we create the object of Derived class using that class we int- the object we access all properties of both classes.

Because of the virtual behaviours 4 bytes extra memory added in the class object for V.PTR

Base class

VTABLE

VPTR	100	200	* 200 → 2000	base::g
	104			base::i
	108			base::f
	112			

Derived class

VTABLE

VPTR	100	200	* 200 → 2000	base::g
	104			base::i
	108			base::f
	112			Derived::i
	116			Derived::d
	120			

class base

{ public :

int i;

float f;

void fun () 11 1000

{ }

virtual void gun () 11 2000

{ }

virtual void sun () 11 3000

{ }

void run () 11 4000

{ }

};

class Derived : Public base

{ public :

int i;

double d;

virtual void fun () 11 5000

{ }

virtual void gun () 11 6000

{ }

virtual void sun () 11 7000

{ }

virtual void run () 11 8000

{ }

};

int main ()

{

 Derived dobj ;
 return 0;

}

- In above Application base class contain 2 virtual functions and 2 normal functions & 2 characteristics.
- Derived class inherit the base class.
- In main method we create the object of base to Derived class.

Base class

		VTABLE	
VPTR	100	*	500
104	500	base::i	2000
108		base::f	3000
112			base::gun
			base::sun

Derived class

		VTABLE	
VPTR	100	*	600
104	500	base::i	2000
108		base::f	3000
112		Derived::i	5000
116		Derived::d	8000
120			Derived::gun
124			Derived::sun

Page No.:	5
Date:	
YOUVA	

6) class base
 {
 public:
 int i;
 float f;
 virtual void func() 111000;
 virtual void sun() 112000;
 };

 class Derived : Public base
 {
 public:
 int i;
 double d;
 virtual void func() 113000;
 virtual void gun() 114000;
 virtual void sun() 115000;
 };

 int main()
{
 Derived dobj;
 return 0;
}

- In the above Application base class having the virtual behaviours gun() & sun() and the characteristics i and f.
- class Derived inherit the base class.
- In main method object of Derived class is created in that object 4 byte extra added because of virtual functions are present in both class.

Base class

VTABLE	
100	
VPTR	200
104	← * 200
108	base::i
112	base::f
1000	base::gun
2000	base::sun

Derived class

VTABLE	
100	
VPTR	500
104	* 500
108	base::i
112	base::f
116	Derived::i
124	Derived::d
1000	base::gun
2000	base::sun
5000	Derived::sun
3000	Derived::fun