

**Topic 3 [Constructor & Destructor in inheritance]:** Write the constructors & destructors

for different types of inheritance are given as follows. Also follow and write the sequence of their execution.

**1.Single inheritance:**

Code:

```
#include<iostream>

using namespace std;

class A
{
private:
    int ax;
public:
    A(int a)
    {
        ax=a;
    }

    int getx()
    {
        return ax;
    }

    ~A()
    {
        cout<<"Destructor of A"<<endl;
    }
}
```

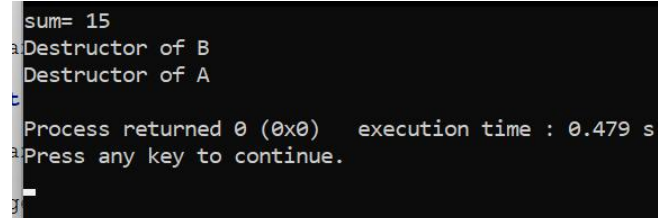
```
};  
  
class B:public A  
{  
private:  
    int bx;  
public:  
    B(int a,int b):A(a)  
    {  
        bx=b;  
  
    }  
    void sum()  
    {  
        cout<<"sum= "<<bx+getx()<<endl;  
    }  
  
    ~B()  
    {  
        cout<<"Destructor of B"<<endl;  
    }  
};  
  
int main()  
{
```

```
B obj(5,10);

obj.sum();

}
```

Output :

A terminal window with a black background and white text. The output shows 'sum= 15', followed by 'Destructor of B' and 'Destructor of A' on separate lines. Below these, it says 'Process returned 0 (0x0) execution time : 0.479 s' and 'Press any key to continue.' with a cursor on the next line.

```
sum= 15
a Destructor of B
a Destructor of A
Process returned 0 (0x0) execution time : 0.479 s
a Press any key to continue.
g
```

## 2.Multilevel inheritance:

Code:

```
#include<iostream>

using namespace std;

class A{

private:

    int ax;

public:

    A(int a){

        ax=a;

    }

    ~A(){

        cout<<"Destructor of A"<<endl;
```

```
}  
  
int getx(){  
    return ax;  
}
```

```
};
```

```
class B:public A{
```

```
private:
```

```
    int bx;
```

```
public:
```

```
    B(int a,int b):A(a){
```

```
        bx=b;
```

```
    }
```

```
    ~B(){
```

```
        cout<<"Destructor of B"<<endl;
```

```
    }
```

```
    int gety(){
```

```
        return bx;
```

```
    }
```

```
};
```

```
class C:public B{
```

```
    int cx;
```

```
public:
```

```

C(int a,int b,int c):B(a,b){

    cx=c;

}

void sum(){

    cout<<"sum= "<<getx()+gety()+cx<<endl;

}

~C(){

    cout<<"Destructor of C"<<endl;

}

};

int main(){

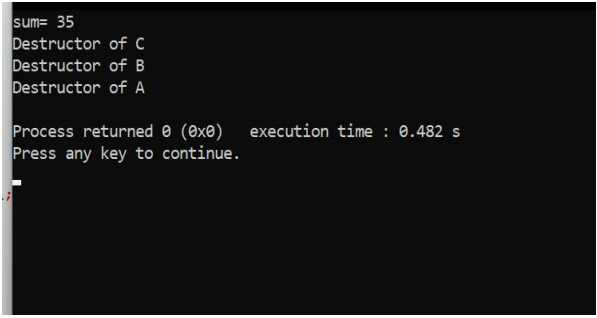
C obj(5,10,20);

obj.sum();

}

```

Output :



```

sum= 35
Destructor of C
Destructor of B
Destructor of A

Process returned 0 (0x0)   execution time : 0.482 s
Press any key to continue.

```

### **3. Multiple inheritance:**

Code:

```
#include<iostream>
```

```
using namespace std;

class A{
private:
    int ax;

public:
    A(int a){
        ax=a;
    }
    ~A(){
        cout<<"Destructor of A"<<endl;
    }
    int getax(){
        return ax;
    }
};
```

```
class B{
private:
    int bx;
public:
    B(int b){
        bx=b;
    }
}
```

```

~B(){
    cout<<"Destructor of B"<<endl;
}

int getbx(){
    return bx;
}

};

class C:public A,public B{
private:
    int cx;
public:
    C(int a,int b,int c):A(a),B(b){
        cx=c;
    }
~C(){
    cout<<"Destructor of C"<<endl;
}

    void sum(){
        cout<<"sum= "<<getax()+getbx()+cx<<endl;
    }

};

int main(){
    C obj(5,10,20);
    obj.sum();
}

```

}

Output :

```
a sum= 35
Destructor of C
Destructor of B
tDestructor of A
a
Process returned 0 (0x0)   execution time : 0.333 s
Press any key to continue.
c
g
r
```