OOP ACTIVITY 3

T. DAMAN

2005839

CSE

Q1.SINGLE INHERITANCE

```
//single inheritance
//T. DAMAN
//2005839
#include<iostream>
using namespace std;
class DAMAN1{
    public:
    DAMAN1(){
        cout<<"constructor of DAMAN1 called "<<endl;
    }
    ~DAMAN1(){
        cout<<"destructor of DAMAN1 called "<<endl;
    }};
class DAMAN2:public DAMAN1{
public:
     DAMAN2(){
        cout<<"constructor of DAMAN2 called "<<endl;
    }
    ~DAMAN2(){
        cout<<"destructor of DAMAN2 called "<<endl;
    }};
int main(){
DAMAN2 r;
return 0;
}
```

OUTPUT:-

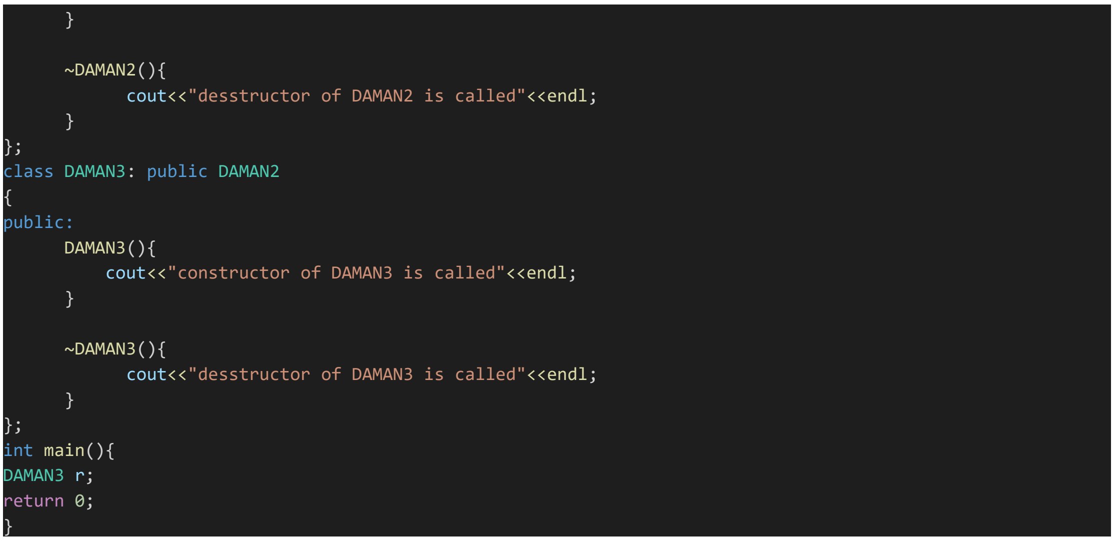
```
constructor of DAMAN1 called
constructor of DAMAN2 called
destructor of DAMAN2 called
destructor of DAMAN1 called
```

Q2.MULTILEVEL INHERITANCE

```
// multi level inheritance
// T.DAMAN
//2005839

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

```
class DAMAN1{
    public:
    DAMAN1(){
        cout<<"constructor of DAMAN1 called "<<endl;
```

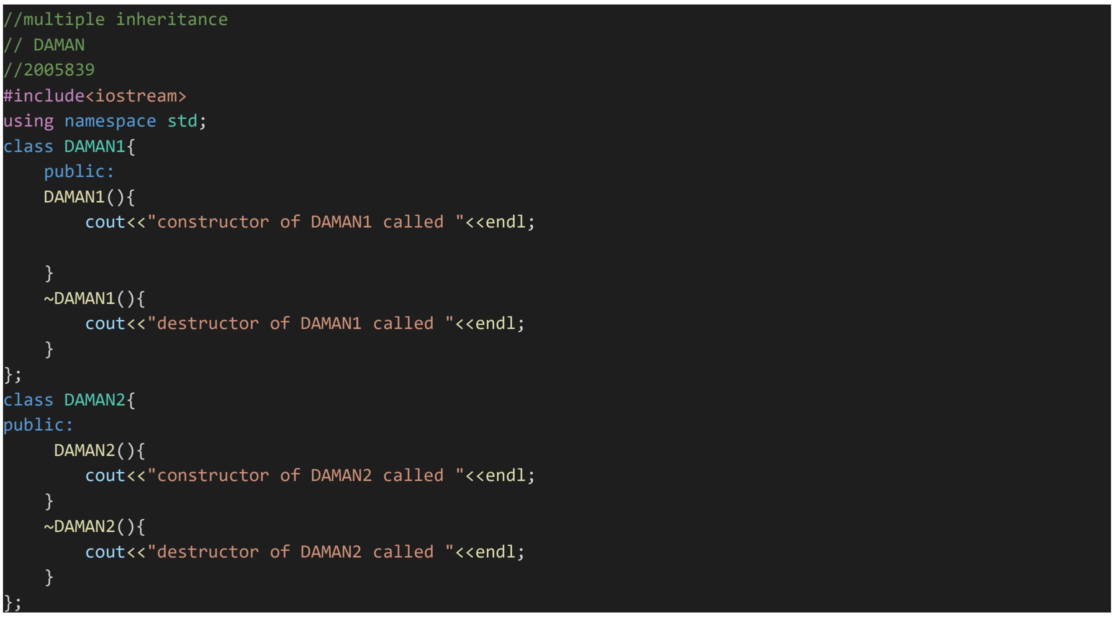
```
    }
    ~DAMAN1(){
        cout<<"destructor of DAMAN1 calLed "<<endl;
    }
};
class DAMAN2:public DAMAN1
{
public:
      DAMAN2(){
          cout<<"constructor of DAMAN2 is called"<<endl;
```



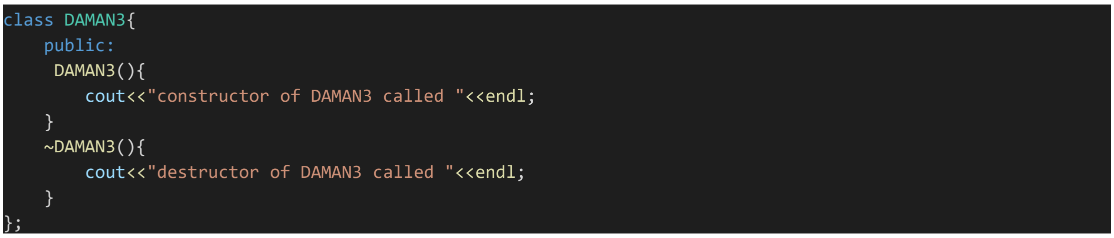
```
      }

      ~DAMAN2(){
            cout<<"desstructor of DAMAN2 is called"<<endl;
      }
};
class DAMAN3: public DAMAN2
{
public:
      DAMAN3(){
          cout<<"constructor of DAMAN3 is called"<<endl;
      }

      ~DAMAN3(){
            cout<<"desstructor of DAMAN3 is called"<<endl;
      }
};
int main(){
DAMAN3 r;
return 0;
}
```

OUTPUT:-
constructor of DAMAN1 called
constructor of DAMAN2 is called
constructor of DAMAN3 is called
desstructor of DAMAN3 is called
desstructor of DAMAN2 is called
destructor of DAMAN1 called

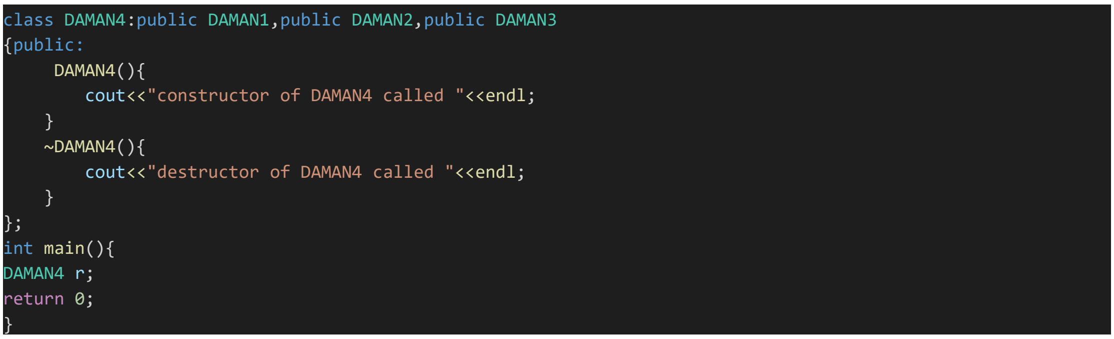
Q3.MULTIPLE INHERITANCE



```
//multiple inheritance
// DAMAN
//2005839
#include<iostream>
using namespace std;
class DAMAN1{
    public:
    DAMAN1(){
        cout<<"constructor of DAMAN1 called "<<endl;

    }
    ~DAMAN1(){
        cout<<"destructor of DAMAN1 called "<<endl;
    }
};
class DAMAN2{
public:
     DAMAN2(){
        cout<<"constructor of DAMAN2 called "<<endl;
    }
    ~DAMAN2(){
        cout<<"destructor of DAMAN2 called "<<endl;
    }
};
```



```
class DAMAN3{
    public:
     DAMAN3(){
        cout<<"constructor of DAMAN3 called "<<endl;
    }
    ~DAMAN3(){
        cout<<"destructor of DAMAN3 called "<<endl;
    }
};
```

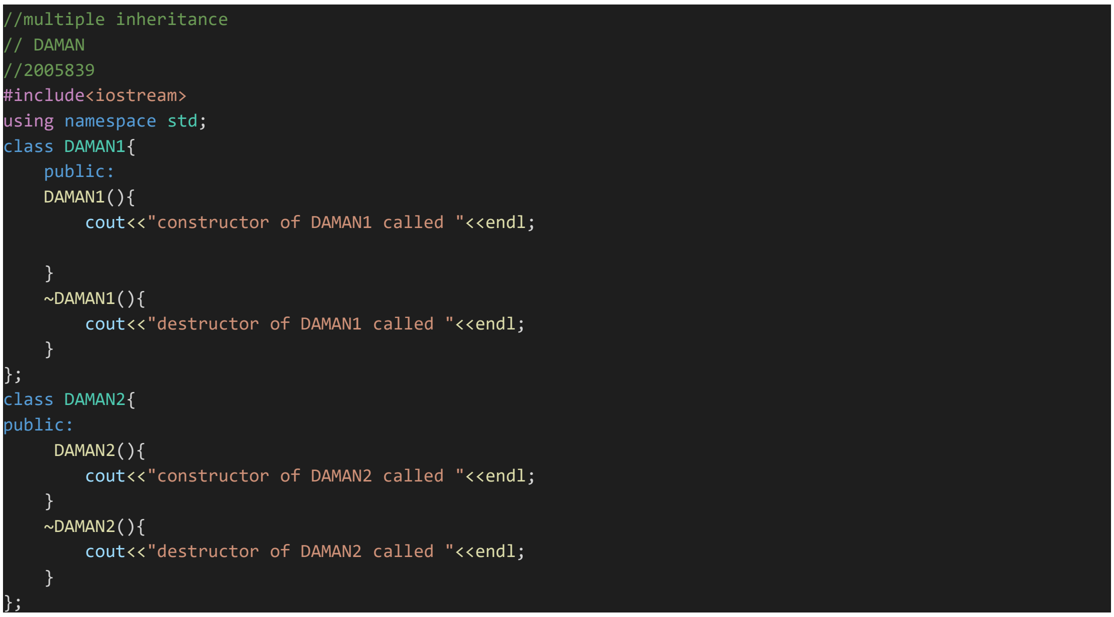


```
class DAMAN4:public DAMAN1,public DAMAN2,public DAMAN3
{public:
     DAMAN4(){
        cout<<"constructor of DAMAN4 called "<<endl;
    }
    ~DAMAN4(){
        cout<<"destructor of DAMAN4 called "<<endl;
    }
};
int main(){
DAMAN4 r;
return 0;
}
```

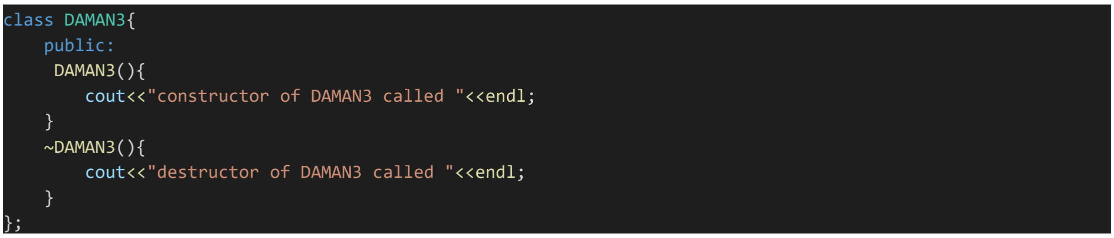
OUTPUT:

```
constructor of DAMAN1 called
constructor of DAMAN2 called
constructor of DAMAN3 called
constructor of DAMAN4 called
destructor of DAMAN4 called
destructor of DAMAN3 called
destructor of DAMAN2 called
destructor of DAMAN1 called
```

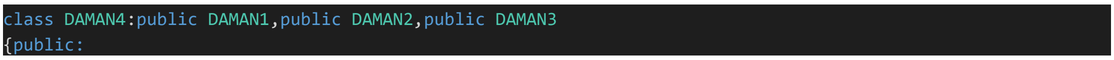
Q4.HYBRID INHERITANCE



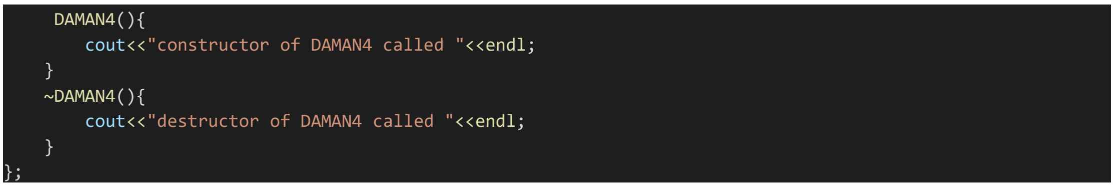
```
//multiple inheritance
// DAMAN
//2005839
#include<iostream>
using namespace std;
class DAMAN1{
    public:
    DAMAN1(){
        cout<<"constructor of DAMAN1 called "<<endl;

    }
    ~DAMAN1(){
        cout<<"destructor of DAMAN1 called "<<endl;
    }
};
class DAMAN2{
public:
     DAMAN2(){
        cout<<"constructor of DAMAN2 called "<<endl;
    }
    ~DAMAN2(){
        cout<<"destructor of DAMAN2 called "<<endl;
    }
};
```



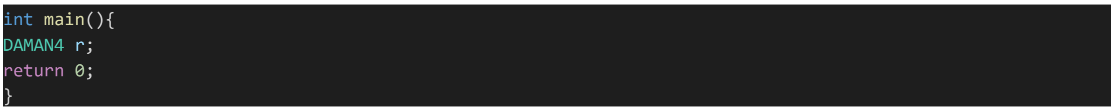
```
class DAMAN3{
    public:
     DAMAN3(){
        cout<<"constructor of DAMAN3 called "<<endl;
    }
    ~DAMAN3(){
        cout<<"destructor of DAMAN3 called "<<endl;
    }
};
```



```
class DAMAN4:public DAMAN1,public DAMAN2,public DAMAN3
{public:
```



```
     DAMAN4(){
        cout<<"constructor of DAMAN4 called "<<endl;
    }
    ~DAMAN4(){
        cout<<"destructor of DAMAN4 called "<<endl;
    }
};
```

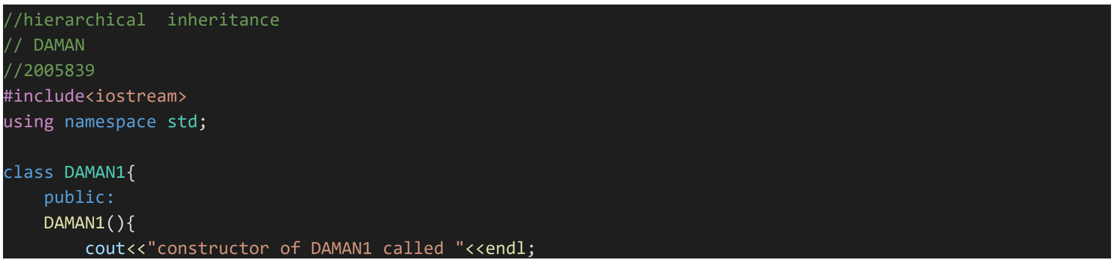


```
int main(){
DAMAN4 r;
return 0;
}
```

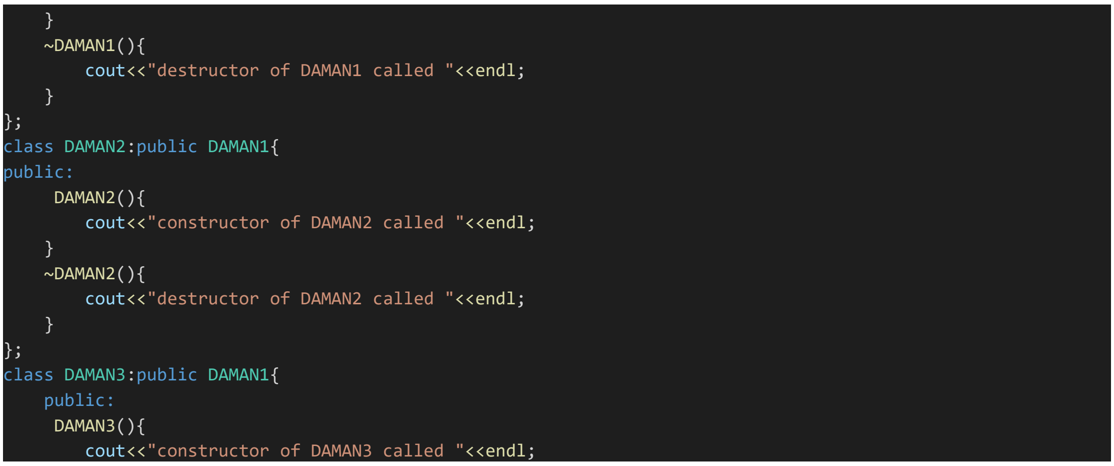
OUTPUT:

```
constructor of DAMAN1 called
constructor of DAMAN2 called
constructor of DAMAN3 called
constructor of DAMAN4 called
destructor of DAMAN4 called
destructor of DAMAN3 called
destructor of DAMAN2 called
destructor of DAMAN1 called
```

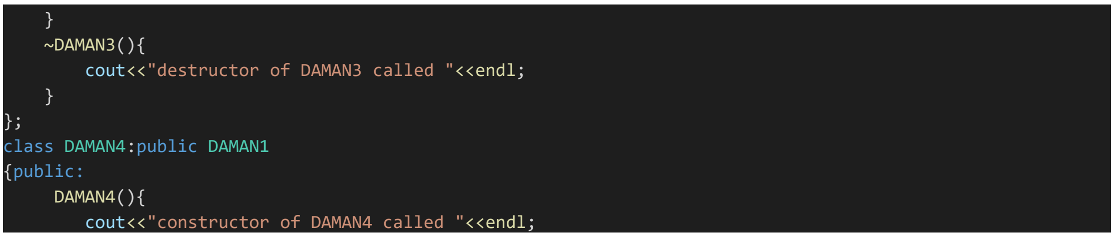
Q5.HIERIRICAL INHERITANCE



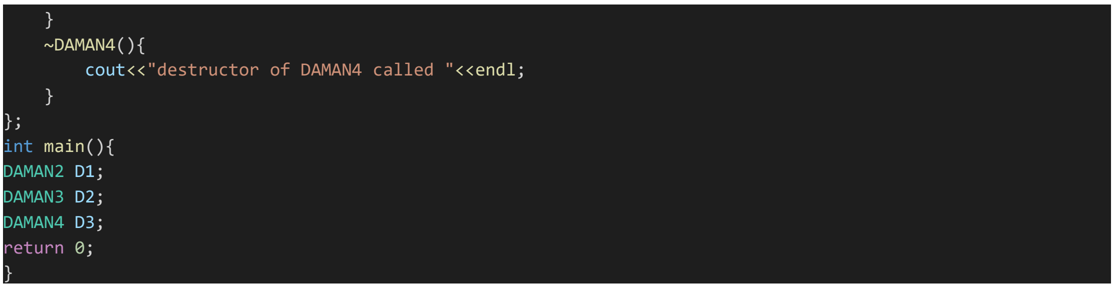
```
//hierarchical  inheritance
// DAMAN
//2005839
#include<iostream>
using namespace std;

class DAMAN1{
    public:
    DAMAN1(){
        cout<<"constructor of DAMAN1 called "<<endl;
```



```
    }
    ~DAMAN1(){
        cout<<"destructor of DAMAN1 called "<<endl;
    }
};
class DAMAN2:public DAMAN1{
public:
     DAMAN2(){
        cout<<"constructor of DAMAN2 called "<<endl;
    }
    ~DAMAN2(){
        cout<<"destructor of DAMAN2 called "<<endl;
    }
};
class DAMAN3:public DAMAN1{
    public:
     DAMAN3(){
        cout<<"constructor of DAMAN3 called "<<endl;
```



```
    }
    ~DAMAN3(){
        cout<<"destructor of DAMAN3 called "<<endl;
    }
};
class DAMAN4:public DAMAN1
{public:
     DAMAN4(){
        cout<<"constructor of DAMAN4 called "<<endl;
```



```
    }
    ~DAMAN4(){
        cout<<"destructor of DAMAN4 called "<<endl;
    }
};
int main(){
DAMAN2 D1;
DAMAN3 D2;
DAMAN4 D3;
return 0;
}
```

OUTPUT:

```
constructor of DAMAN1 called
constructor of DAMAN2 called
constructor of DAMAN1 called
constructor of DAMAN3 called
constructor of DAMAN1 called
constructor of DAMAN4 called
destructor of DAMAN4 called
destructor of DAMAN1 called
destructor of DAMAN3 called
destructor of DAMAN1 called
destructor of DAMAN2 called
destructor of DAMAN1 called
```