Day 09

26 March 2025 08:31 AM

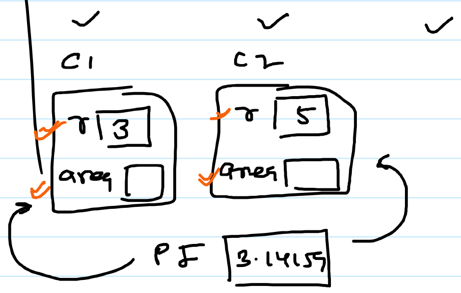
static

Instance member

static

```
class circle
{
    int r;
    float area;
    static float PI;
public:
        void cal_area()
        {
            area = PI*r*r;
        }
};
float circle::PI = 3.14159;
```

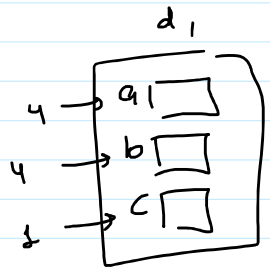
```
int main()
{
    circle c1=3;
    circle c2=5;
```

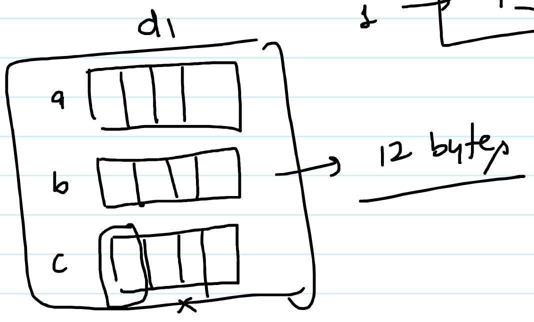


```
#include<iostream>
using namespace std;
class circle{
    int r;
    float area;
    static float PI;
    public:
        circle(int r1=0)
        {
            r=r1;
        }
        void cal_area()
        {
            area = PI*r*r;
        }
        void output()
        {
            cout<<"Area of circle = "<<area<<endl;
        }
};
float circle::PI=3.14159;
int main()
{
    circle c1=5,c2=7,c3=15;
    c1.cal_area();
    c2.cal_area();
    c3.cal_area();
    c1.output();
    c2.output();
    c3.output();
    return 0;
}
```

```
struct data
{
        int   a;
        float  b;
        char  c;
};
```

```
data  d1;
sizeof (d1) →
```





```
struct   data
{
        int q;
        double r;
        char  s;
        char  p;
};
```

```
data  d1;
sizeof (d1);  →
```

```
#include<stdio.h>
struct data{
    char x;
    float a;
    int b;
    char c;
};
int main()
{

   printf("%d",sizeof(struct data));

}
```

Types of member func :→

① Instance

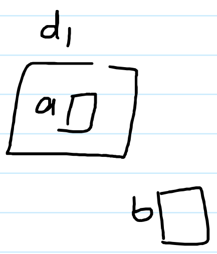
② static

```
class data
{
    int a;                    <- instance variable
    static int b;             <- static variable

  public :
                                          this [4d1]  local
                                          [10]   [20]
        void f1 (int a, int b);
        {
            this -> a = 10 ;   - ins
            data :: b = 20 ;   <- static
        }

        ~~that~~ static void f2 ()
        {
            data :: b = 20 ;
            a = 10 <- X
        }
};
int data :: b;
```

instance func

static func

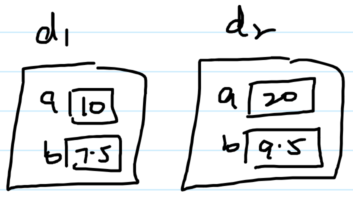
```
int main ()
{
    data d1;
    d1.f1 (10, 20);      4d1   10   20
    [4d1]
    d1.f2 ();
    data :: f2 ();
```



✓ const func :->

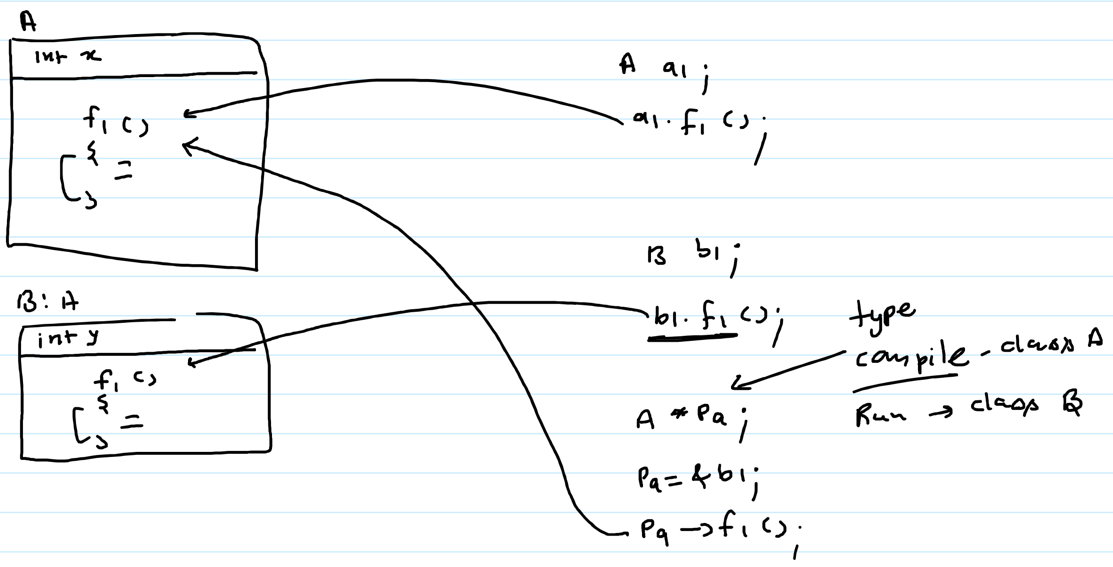
```
class data
{
        int a;
        float b;
    public :
        data (int a1 , float b1)
        {
            a = a1;
            b = b1;
        }

        void output () const
        {
            cout<<a << b;
            ~~a = 10 ;~~
        }
};
```

```
int main ()
{
    data d1 (10, 7.5)
    const data d2 (20, 9.5);
    d1.output();
    d2.output ()
```



```
#include<iostream>
using namespace std;
class data{
    int a;
    mutable float b;
    public:
        data(int a1, float b1)
        {
            a=a1;
            b=b1;
        }
        void output() const
        {
            //a=10;         error
            b=200;
            cout<<a<<" "<<b<<endl;
        }
};
int main()
{
    data d1(10,7.5);
    const data d2(20,9.5);
    d1.output();
    d2.output();
    return 0;
}
```

virtual func :->

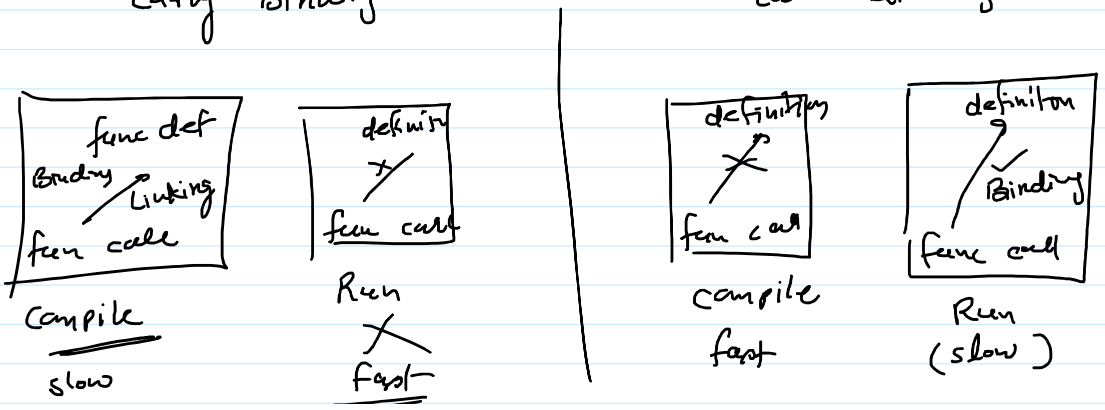


```
A a1;
a1.f1();

B b1;
b1.f1();

A *Pa;
Pa=&b1;
Pa->f1();
```

Binding

→ Compile time (default)
or
Static Binding
or
Early Binding

Run time
or
Dynamic Binding
or
Late Binding



Compile

slow

~~fast~~

fast

Run

(slow)

```
#include<iostream>
using namespace std;
class A
{
    public:
        virtual void f1()
        {
            cout<<"F1 of class A\n";
        }
};
class B:public A{
    public:
        void f1()
        {
            cout<<"F1 of class B\n";
        }
};
int main()
{
    A *p;
    p=new B;
    p->f1();
    return 0;
}
```

Output
F1 of class B

```
include<iostream>
using namespace std;
class Mobile{
    public:
        virtual void network()
        {
            cout<<"Mobile class";
        }
};
class Airtel : public Mobile{
    public:
    void network()
    {
        cout<<"Airtel";
    }
};
class BSNL:public Mobile{
    public:
    void network()
    {
        cout<<"BSNL";
    }
};
int main()
{
    Mobile *p;
    int n;
    cout<<"1. Airtel\n2.BSNL\n";
    cin>>n;
    if(n==1)
        p=new Airtel;
    else
        p=new BSNL;
    p->network();
    return 0;
}
```

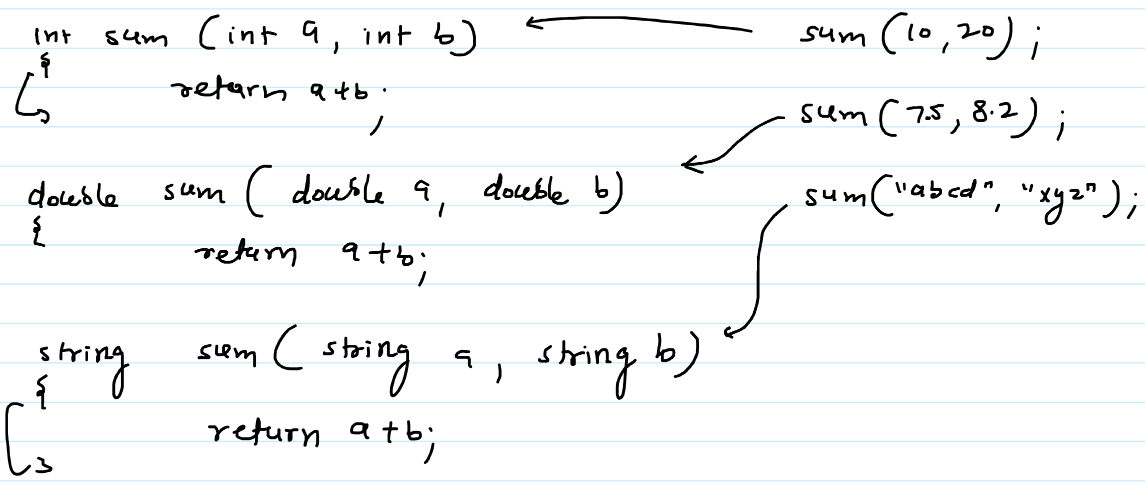
```
#include<iostream>
using namespace std;
```

```
class Mobile{        //abstract class
    public:
        virtual void network()=0;       //pure virtual function
};
class Airtel : public Mobile{
    public:
    void network()
    {
        cout<<"Airtel";
    }
};
class BSNL:public Mobile{
    public:
    void network()
    {
        cout<<"BSNL";
    }
};
int main()
{
    //Mobile m1;        error
    Mobile *p;
    int n;
    cout<<"1. Airtel\n2.BSNL\n";
    cin>>n;
    if(n==1)
        p=new Airtel;
    else
        p=new BSNL;
    p->network();
    return 0;
}
```

Output

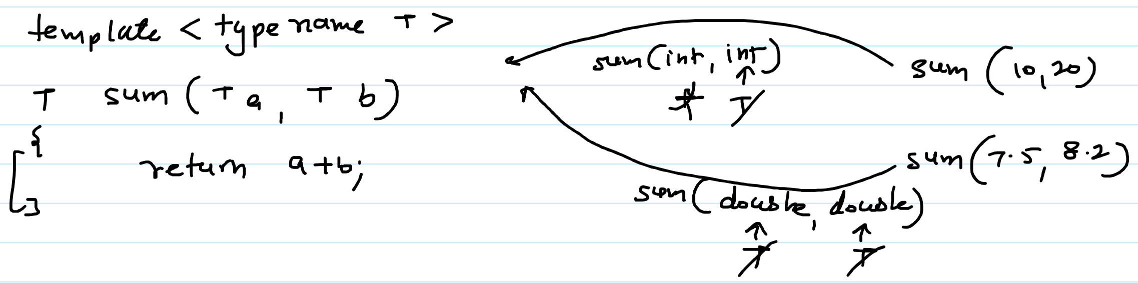
```
1. Airtel
2.BSNL
1
Airtel
```

Template :->



```
int sum (int a, int b)
{
        return a+b;
}

double sum ( double a, double b)
{
        return a+b;

string sum ( string a, string b)
{
        return a+b;
}
```

```
sum(10,20);
sum(7.5, 8.2);
sum("abcd", "xyz");
```



```
template < typename T >
T sum ( T a, T b)
{
        return a+b;
}
```

```
#include<iostream>
#include<string>
using namespace std;
template<typename T>
T sum(T a, T b)
{
    return a+b;
}
```

```
int main()
{
    cout<<sum(10,20)<<endl;
    cout<<sum(7.5,8.2)<<endl;
    string s1="abcd",s2="xyz";
    cout<<sum(s1,s2)<<endl;
    return 0;
}
```

```
#include<iostream>
#include<string>
using namespace std;
template<typename T1,typename T2>
double sum(T1 a, T2 b)
{
    cout<<"Template ";
    return a+b;
}
int sum(int a, int b)
{
    cout<<"Int ";
    return a+b;
}
int main()
{
    cout<<sum(10,20.5)<<endl;
    cout<<sum(7.5f,8)<<endl;
    cout<<sum(7,8)<<endl;
    return 0;
}
```

```
#include<iostream>
using namespace std;
template<typename T>
class Array{
     T arr[10];
     int n;
     public:
          void input()
          {
               cout<<"Enter number of
elements:";
               cin>>n;
               //input
               for(int i=0;i<n;i++)
               {
                    cout<<"Enter value of
"<<i+1<<" element:";
                    cin>>arr[i];
               }
          }
          void output()
```

```
        {
            for(int i=0;i<n;i++)
            {
                cout<<arr[i]<<" ";
            }
            cout<<endl;
        }
};
int main()
{
    Array<int> a1;
    Array<float> a2;
    a1.input();
    a2.input();
    a1.output();
    a2.output();
    return 0;
}
```