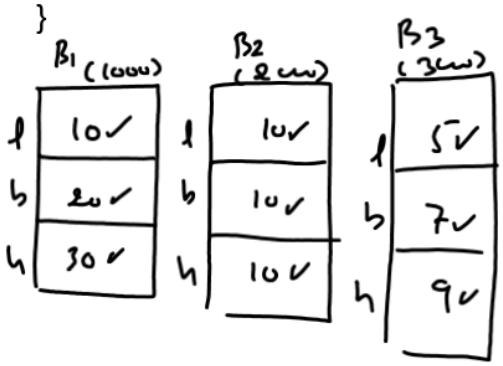
```
#include <iostream.h>
#include <conio.h>
class Box
{
  int l,b,h;
public:
        Box();
        Box(int);
        Box(int,int,int);
        void show();
};
Box::Box()
{
   cout<<"Enter l,b,h:";
   cin>>l>>b>>h;
}
Box::Box(int s)
{
   l=b=h=s;
}
```

```
Box::Box(int i,int j,int k)
{
   l=i;
   b=j;
   h=k;
}
void Box::show()
{
   cout<<l<<","<<b<<","<<h<<endl;
}
```



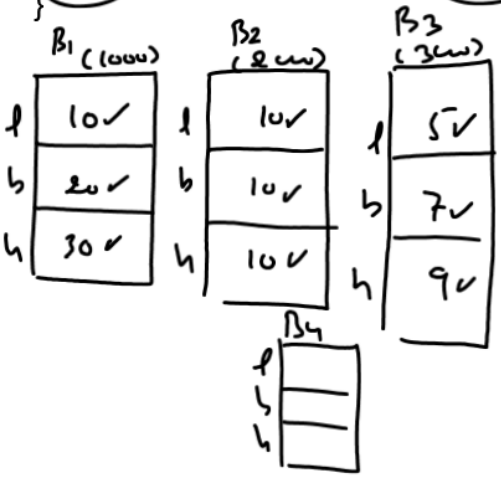

```
int main()
{
   clrscr();
   Box B1;
   Box B2(10);
   Box B3(5,7,9);

   B1.show();
   B2.show();
   B3.show();


   getch();
    return 0;
}
```

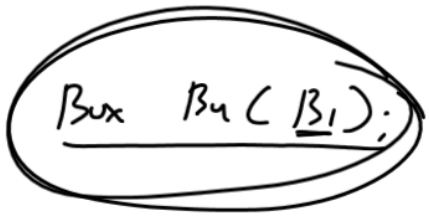
```
#include <iostream.h>
#include <conio.h>
class Box
{
  int l,b,h;
public:
        Box();
        Box(int);
        Box(int,int,int);
        void show();
};
Box::Box()
{
   cout<<"Enter l,b,h:";
   cin>>l>>b>>h;
}
Box::Box(int s)
{
   l=b=h=s;
}
```

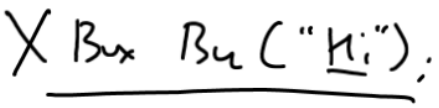
```
Box::Box(int i,int j,int k)
{
   l=i;
   b=j;
   h=k;
}
void Box::show()
{
   cout<<l<<","<<b<<","<<h<<endl;
}
```




```
int main()
{
   clrscr();
   Box B1;
   Box B2(10);
   Box B3(5,7,9);
   Box B4;
   B4=B1;
   B1.show();
   B2.show();
   B3.show();


   getch();
    return 0;
}
```

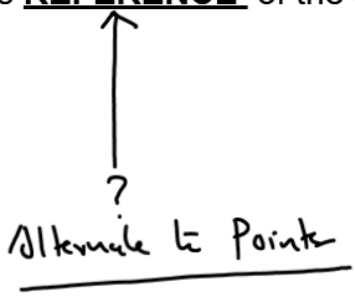




Copy constructor
==========

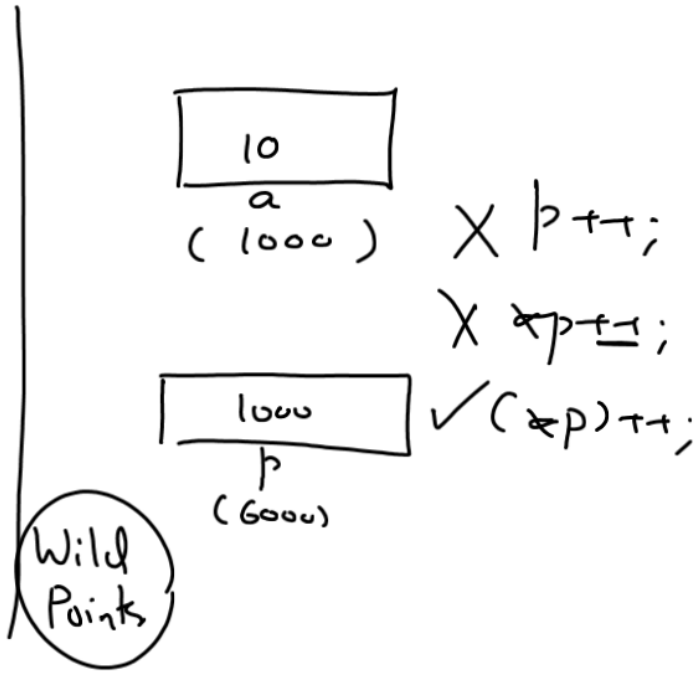
1. A copy constructor in C++ is a SPECIAL CONSTRUCTOR of the class which is used for copying an **existing object** in a **newly created object**

2. To create a copy constructor in the class, we have to declare a constructor which accepts **REFERENCE** of the object of its own class as argument.



C++

```
int main( )
{
    int a = 10;
    int *p;
    p = &a;
    cout << *p;
    return 0;
}
```



<data type> & <reg> = var;

```
int a = 10;
int &p = a;
cout << p; → 10
p++;
cout << a; → 11
```

Your next clas will be 15th Sep at 7pm.