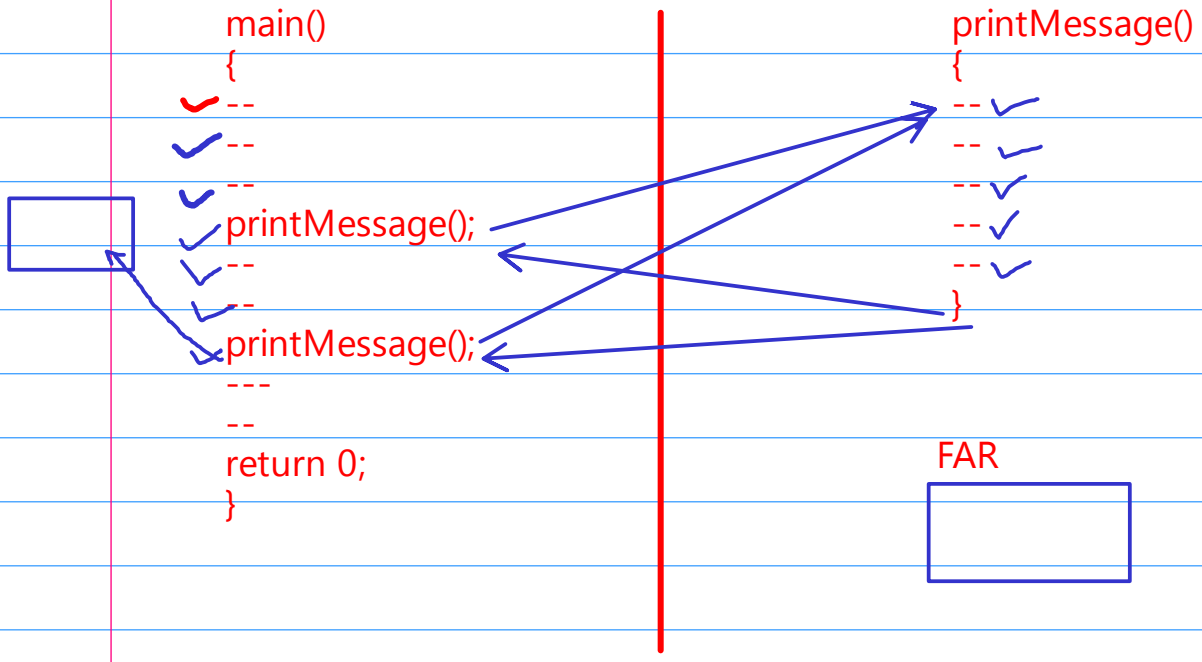
DAC cpp =100%
preCAT CPP =35%
DESD cpp = 70%

CPP= c+oop



#

void printValue(int a) =>printValue@int

void printValue(int a,int b) =>printValue@int,int

void printValue(char a) =>printValue@char

void printValue(int a,char b)=>printValue@int,char

void printValue(char a,int b)=>printValue@char,int

1 / 0 int => 4 => 32 1
bool => 1 => 8

char => 1 => 8 =>256
wchar_t => 16 => 2 =>65536

book

pages
auther
price
pub
sal
roll_no

time

hr
min
sec

print()

accept()

this=4000

{
this-> hr

t1

hr	min	sec
11	22	33

2000

t2

hr	min	sec
10	3	20

3000

t3

hr	min	sec
8	5	33

4000

t1.accpet()

t2.accept()

t3.accept()

c++ => this
java=> this
c# => this
python => self

Structure in C

```
struct time {
   int hr, min, sec;
};
void accept( struct time *p) {
   scanf("%d:%d:%d", &p→hr,
   &p→min, &p→sec);
}
Main()
{
struct time t;
accept(&t);
 }
```

gb

7000

7000->hr

hr min sec

11

7000

7000

class in C++

```
class time {
   int hr, min, sec;
void accept(){
   scanf("%d:%d:%d",&hr,  &min,
   &sec);
}
}; //end of class
Main()
{
time t;
t.accept();
 }
```

mf

const

time * this

&this->hr

hr min sec

12

9000

current obj
calling obj

basic

app

req

int n1

n1=12

n1=55

int& ref = n1

ref=66

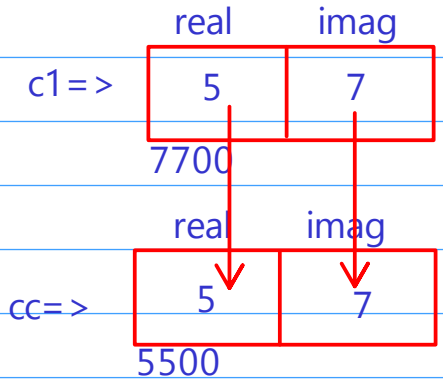
n1=> 66
ref => 66

n1 ref

66

5500

n=> int type vari
int* p=> int pointer type vari
int& r=> int ref type vari



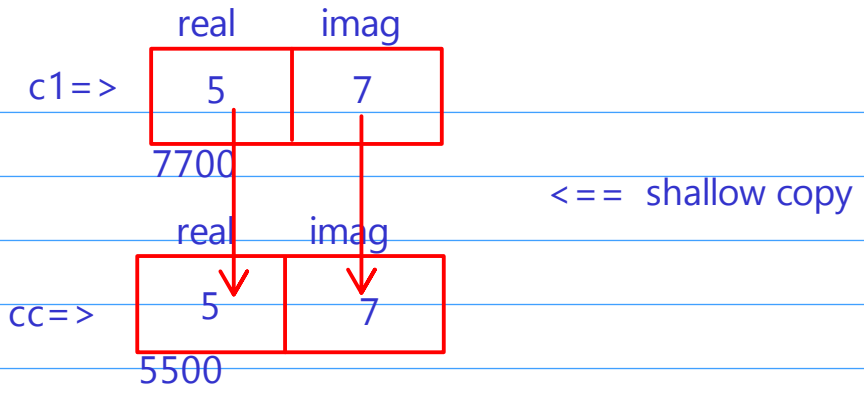

```
complex
{
pub:
sum( compex& c2)
{
        this->real  //c1
}


}
```

```
main()
{
        complex c1(7,6)
        complex c2(3,2)
        complex c3

//c1.real+c2.real

c1.sum(c2)

}
```

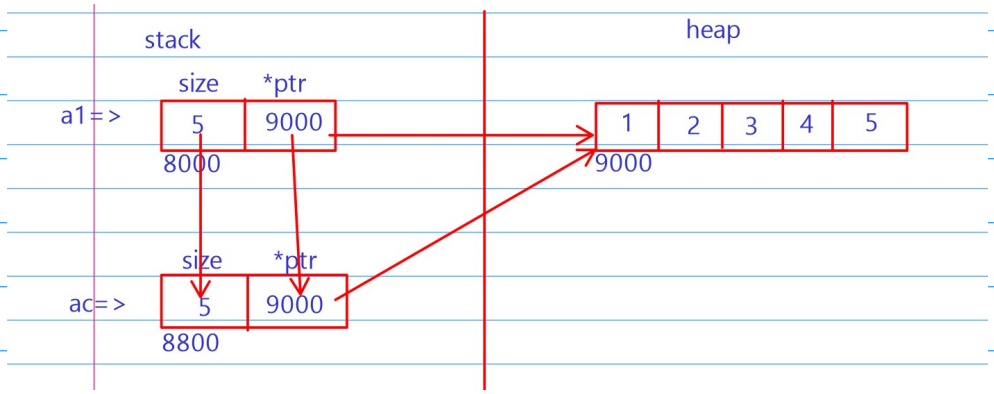
time
hr
min
sec

complex
real
imag

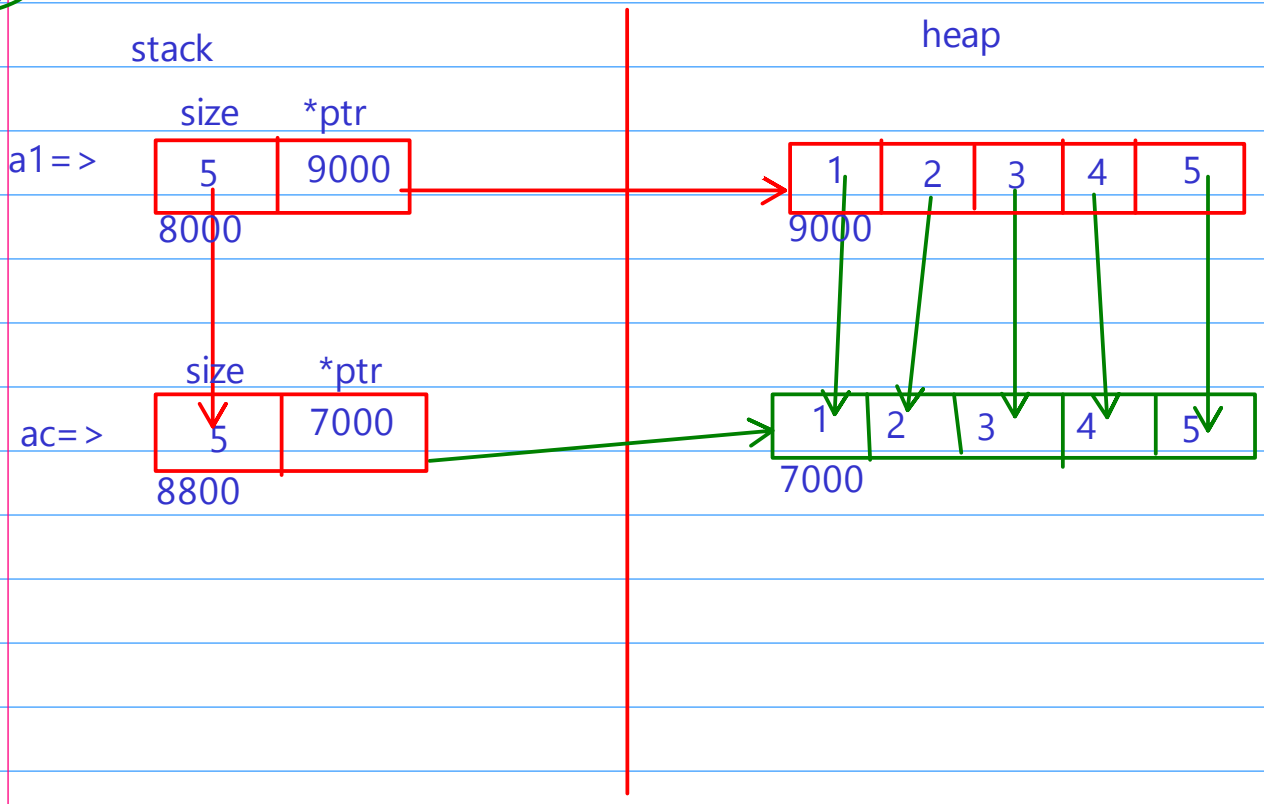
car
price
engine



shallow copy



Deep copy



Abstraction

Encapsulation

```
main()    {
 printf("---",---);

}
```

```
printf()
{
--
--
--
--
--
-
--
}
```

```
main()
{
account a1;
a1.


}
```

```
class account
{
private:
no.
name
email
bal
pan
fun1()
intr()
pub:
withdraw()
deposite()
```

4

DM=2
MF=2

emp is-a person

DM=4
MF=4

person
{
name
age

printPerson() //2
accept() //2
}

emp : public person
{

empid
sal

printEmp()//4
updateName()
accept() //4
}

person p1

name	age
rajesh	24

p1

1100

person
*pptr

2200

name	age	empid	sal
rajesh	24	124	25k

2200

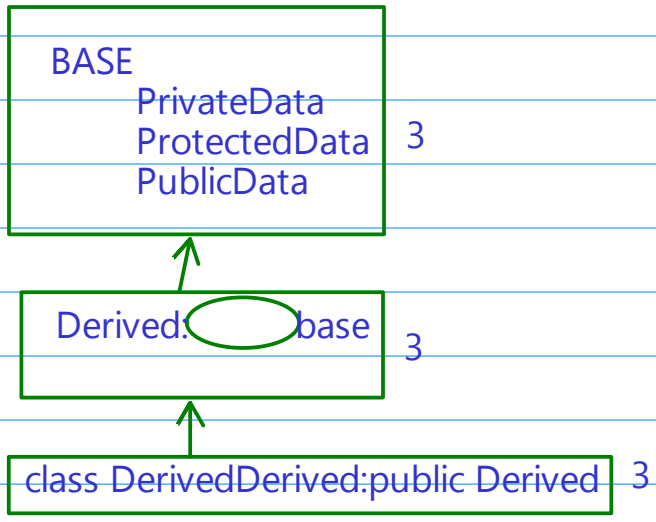
person p2

rajesh	24

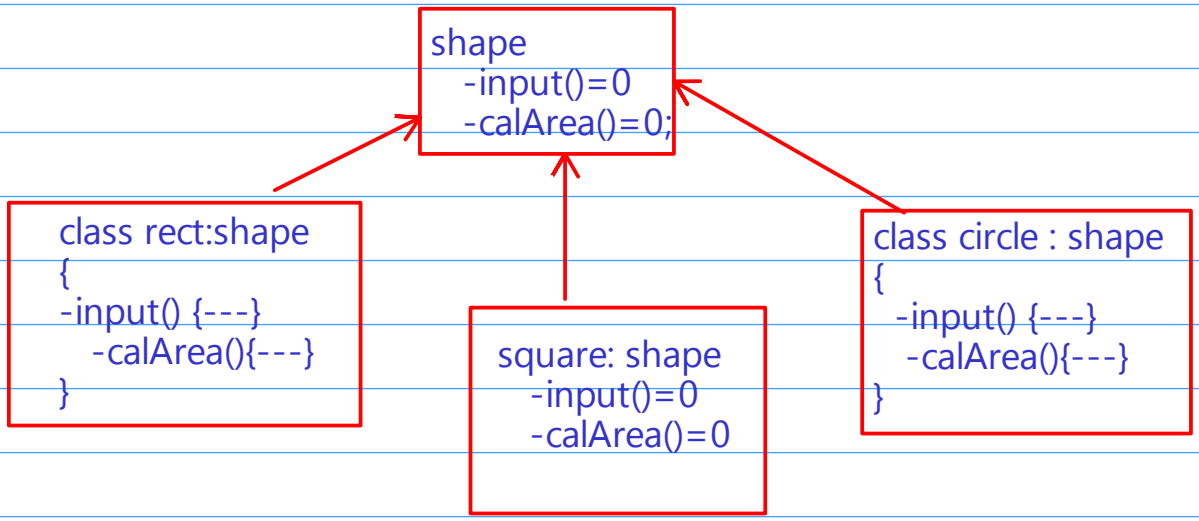
3300

char ch
float f
f=ch

p2=e1



class other
Derived d



	simple	virtual	virtual =0
base	printPerson()	accept() //2	calArea()
derived	fully implemented	partially accept() //4	fully unimplemented