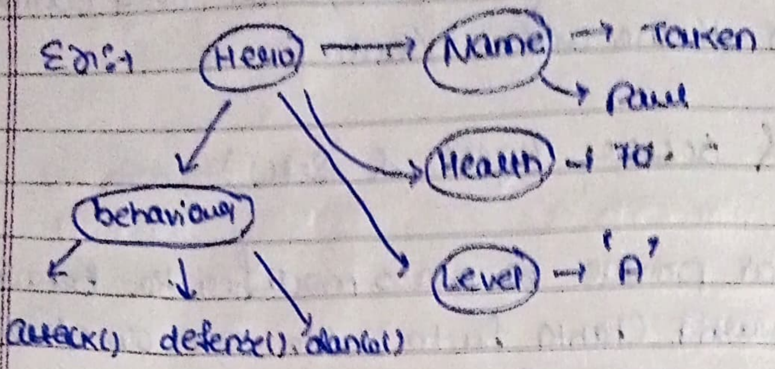
OOPS

Object → Entity → State/properties
→ behaviour

Ex:-



Class → user defined datatype like others data type int, string, char this class is defined by user

Object

empty class ke case mai object ko 1 bit allocate ho jayegi;

→ agar mere object ko access karna hai Hero ~~ko~~ ki properties so it can be done by using (.) Operator

Accessmodifier

→ Default() :→ agar kuch nhi hai to default hota hai private in ~~C++~~ C++;

→ private :→ mith mai is class ka moniter hu meri is class mai chalegi

→ Public :→ Can be access by anywhere & everyone (he is backbencher of class life like ~~the~~ king size)

→ Protected :→ protected is similar to private but is can be access ~~to~~ by with the help of friend class

The difference is that the class member declared as proteced can be accessed by any Subclass (derived class) of the class as well.

→ mera Dost ko janta hai / mai iska aadmi hu pr kudne wala insaan.

getter & setter (get() & set())

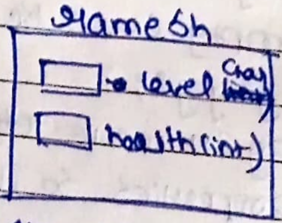
agar mai private ~~ko~~ Access modifier ko behaviour ko use karna chanta hu to usko liye do chize hai mere pass getter & setter.

getter → to get data / name of private
setter → to set ~~a~~ new data to that name

||

BTS of Hero Ramesh

padding, greading alignment

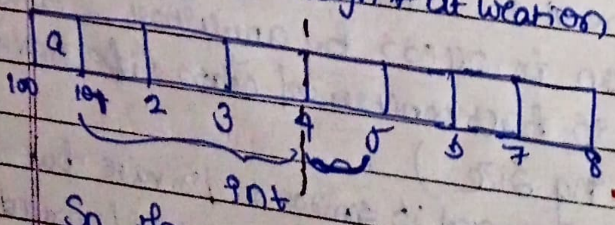


If my cpu is I can acces
word size { 32 bits → 4 bytes (mostly cpu)
64 bits → 8 bytes
In one cpu cycle

so size iska level → bits
health 4bits
Total 5bits hona chiye
↓
but hota 8 bits hai
↓
here structure padding & alignment concept arize

So

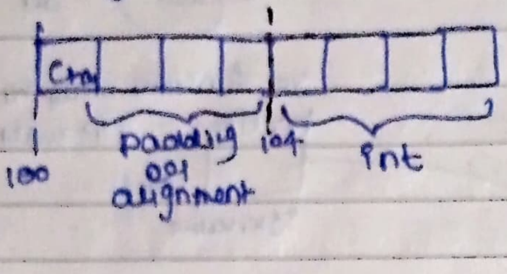
let's suppose
for char a → we asign it at location



So for receiving all

This integer bytes I have to run Two cpu cycle for integer only, for solving those kind off problem we do not assigning in cpu only location multiple of 4

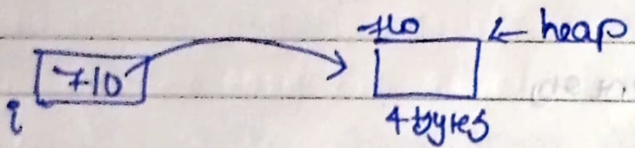
So we gonna do



→ This is gonna increase my cpu speed

// → dynamic Allocation

int * i = new int;



i pointing to 710 Memory present in heap

Similarly we can also point in oops

Hero * h = new Hero;

we can access data member by h → gethealth()
h → sethealth()

// Constructor

Hero ramesh   but BTS
↳ there is an default constructor invoked ramesh.Hero()

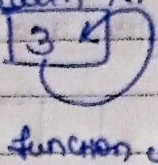
→ automatically invoked at the time of object creation.

→ we can create a own constructor also have Same name of Class

Parameterised constructor

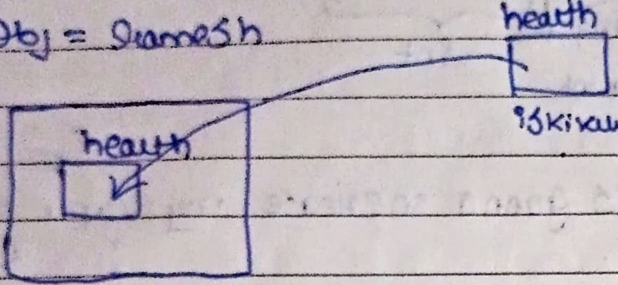
```
Hero (int Health) {
    Health = Health;
}
```

} ye abhi esa kar rahi hai ki health ki value khud health mai dal di



But we want
Obj = Ramesh

ye function wala hai health i/p wala



iski value

isme aa jaye

* <u>this</u> Keyword
↓
current obj
↓
address ko store karta hai
↓
<u>this</u> keyword mai

agar maine ek bhi constructure khud bana liya to default constructor dead ho jayega.

* copy constructor

→ maine ek constructor banaya usko dusre mai pass kardiya

* mistake in it

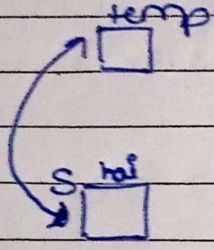
→ mai khud ka constructor create kar raha hai

```
Hero (Hero temp) {
    this-> health = health
}

int main()
  Hero R(S)
```

mai isko call kiya
pass by value kiya hai
S ki nayi copy bani hai

when we do pass by value mai copy banti hai



S ~~ki valu~~ copy hua temp mai
temp ko pass by value to copy constructor call hua fir usne temp ki value lane ki kosis kari ese hi ye loop mai chalega.

↓

isko solve karna ke liye mai pass by refrence karunga & use karunga