

feature 1 : we can directly import struct into function
feature 2 : separate object will be created

Ds

heap
+ week D?
fune

Structure as Parameter

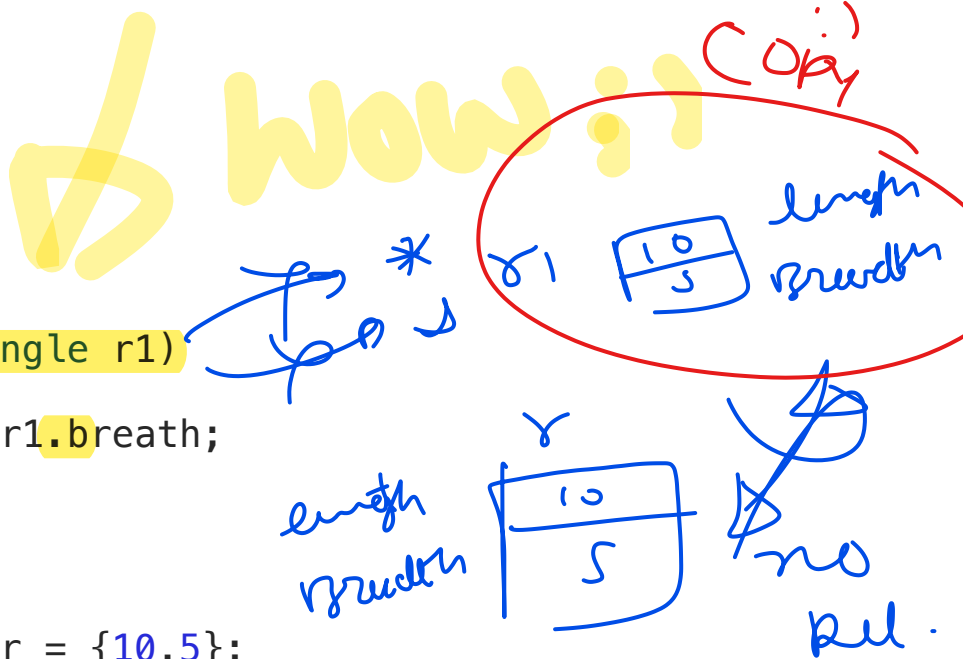
- If we are sending structure as a parameter to a structure it may be call by value or call by address

call by value

- Example

```
int area(struct rectangle r1)
{
    return r1.length*r1.breath;
}

int main()
{
    struct rectangle r = {10,5};
    printf("%d", area(r));
}
```



- A separate object will be created in call by value method and everything will be copied in the corresponding members this is one of the benefit of structure as parameter
- If you are making changes to formal parameter it will not effect the actual parameter

call by reference

- Example

```
int area(struct rectangle &r1)
{
    return r1.length*r1.breath;
}

int main()
{
    struct rectangle r = {10,5};
    printf("%d", area(r));
}
```



- The only change we need to do for call by reference is adding & in the parameter
- Here the new object is not created but the **same object is called r1** also
- Thus new changes in the values will effect the actual parameters



call by address

- Example

```
Void changeLength(struct rectangle *p , int l)
```

```
{  
    p -> length = l;  
}
```

```
int main()    make main func as manager who just assign orders to other functions and not doing work by  
{            himself  
    struct rectangle r = {10,5};  
    ChangeLength(&r, 20);  
}
```

- If you want some function to change the actual parameter then it must be done by call by address or call by reference

```
void fun(Struct Test t1){  
    t1.A[0]=10; //won't change original  
}
```

- It is possible to send array as a parameter in pass by value **only** if its inside the **structure** , if its just array passing then its not possible by pass by value

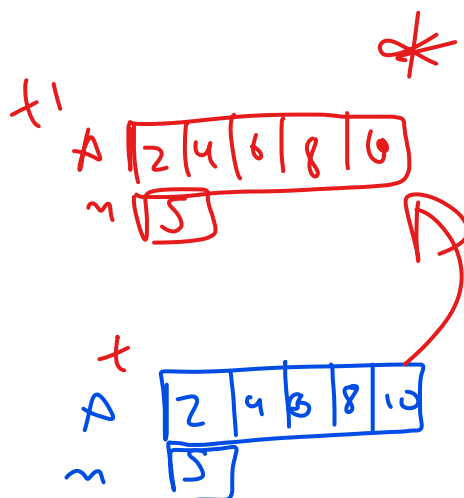
can we send struct by call by value ? yes

can we send array by call by value ? no ,only by call by address

juggadarray inside struct :

```
struct Test{  
    int A[5];  
    int n;  
};
```

```
struct Test t={{2,4,6,8,10},5};  
fun(t)
```



"to avoid too much time consuming to copy each stuff of array..."

thats whay compiler dont allow array call by value"

but inside of structure,they allow it