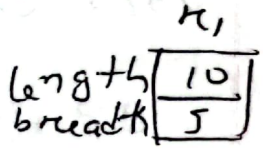
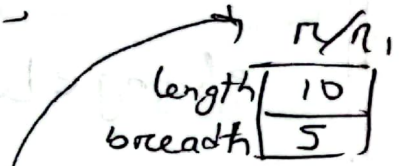
Structure as Parameter

call by value

```
struct Rectangle {
  int length;
  int breadth;
};

int area (struct Rectangle r1) {
  return r1.length * r1.breadth;
}

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

int area (struct Rectangle &r1) {
   return r1.length * r1.breadth;
}
```



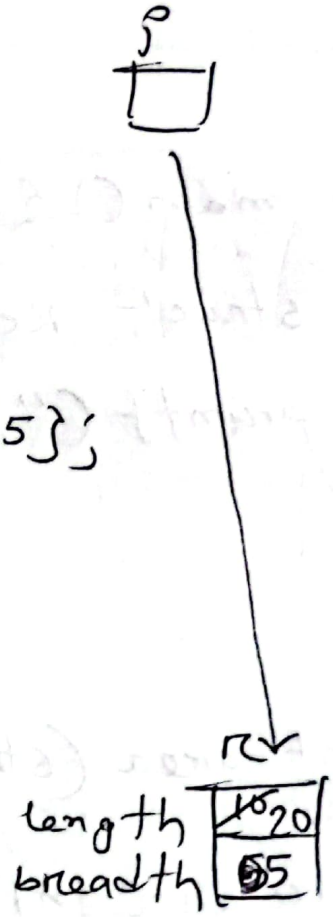


Call by reference

Call by Address

```
struct Rectangle {
  int length;
  int breadth;
};

void changeLength(struct Rectangle *p, int l){
  p->length = l;
}

int main(){
  struct Rectangle r = {10, 5};
  changeLength(&r, 20);
  cout << r.length << endl;
}
```

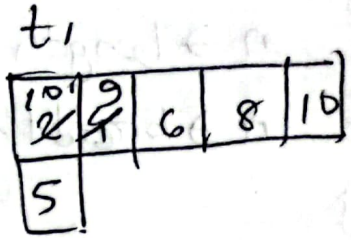
20



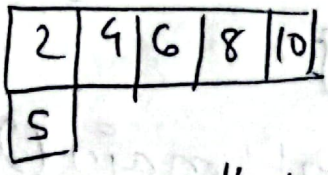
Array inside a structure:

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

void func(struct Test t1) {
  t1.A[0] = 10;
  t1.A[1] = 9;
}

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

array is called by value



array will be copied but can't be modified as it called by value



array won't be modified!!