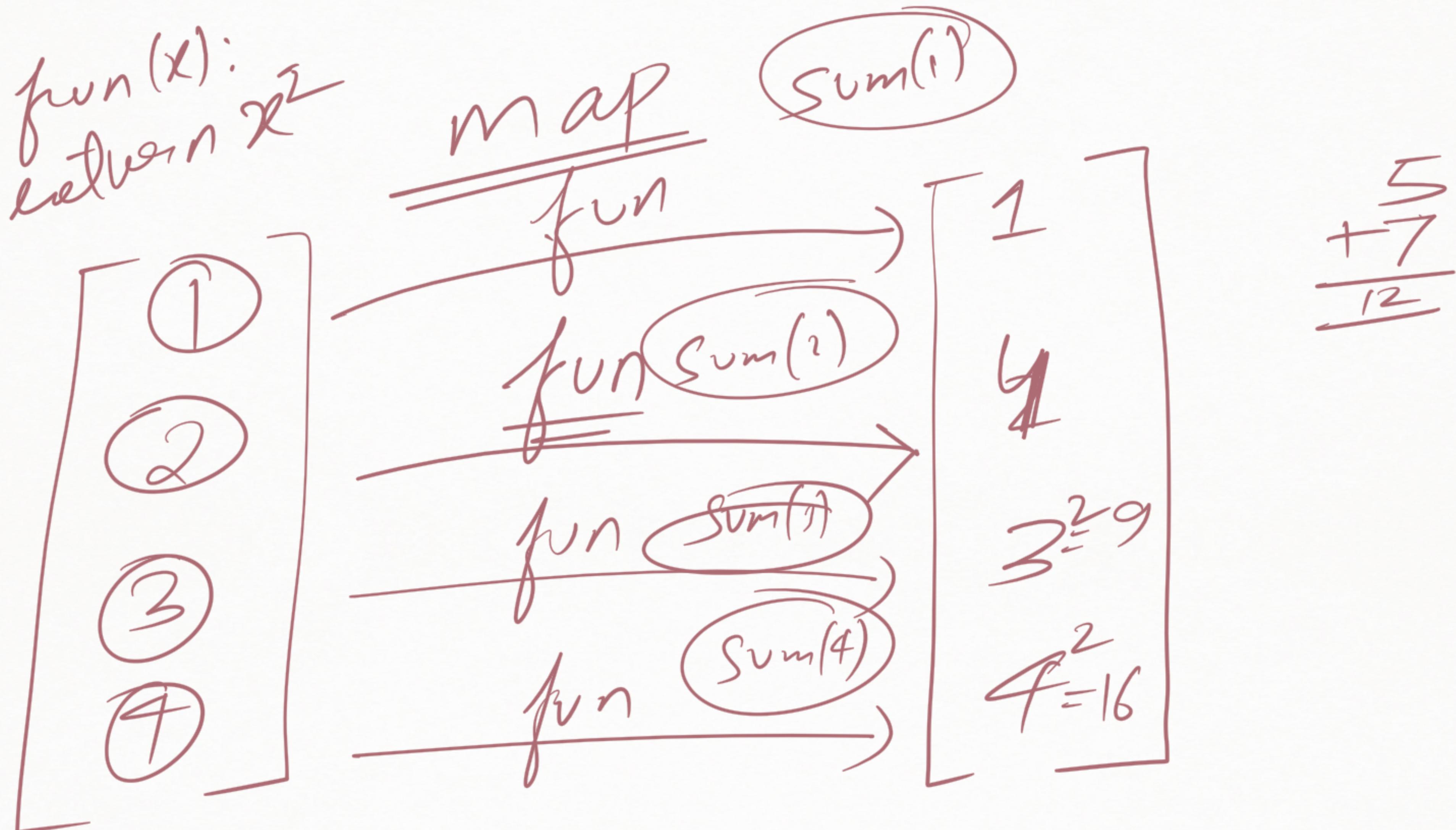
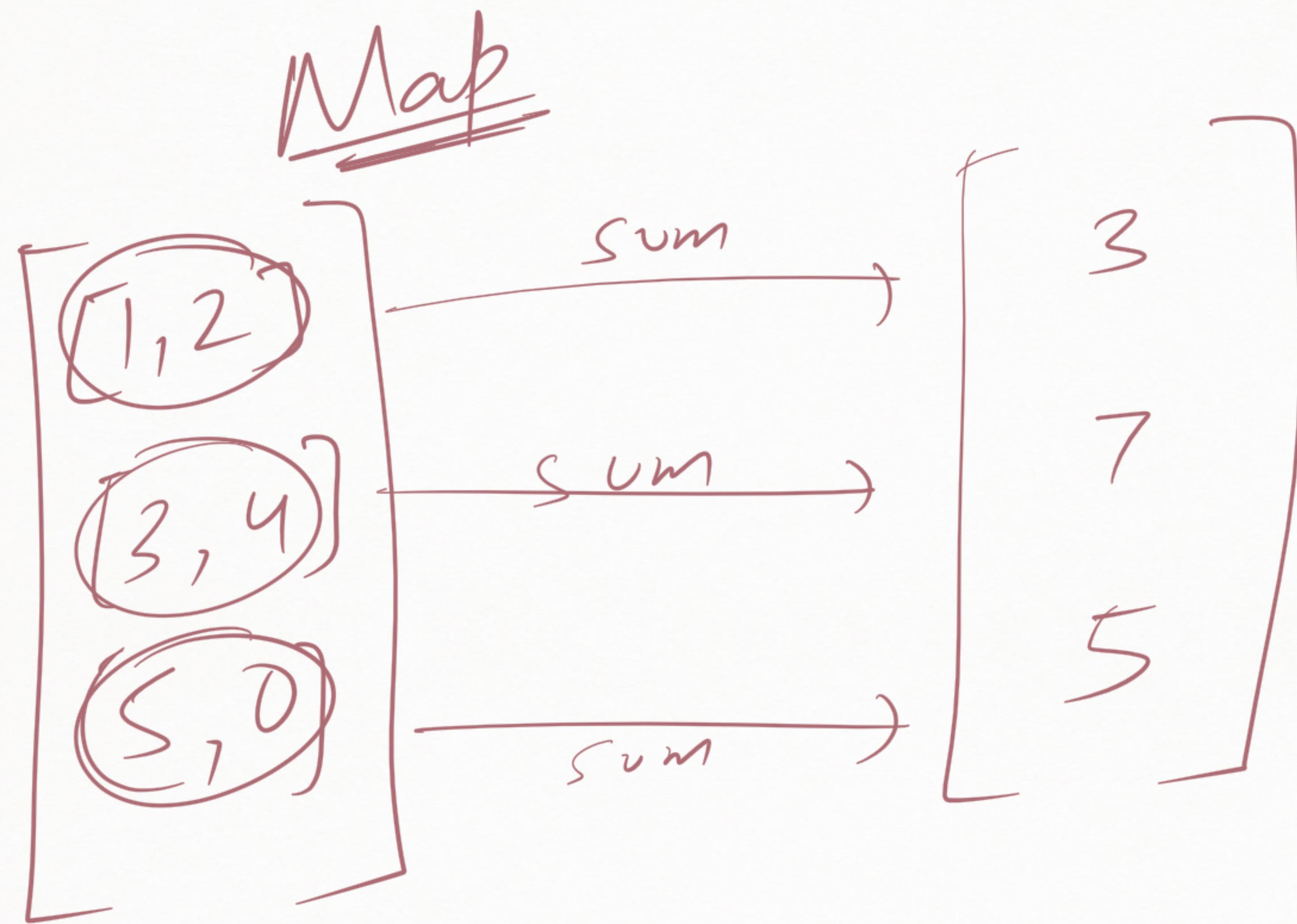
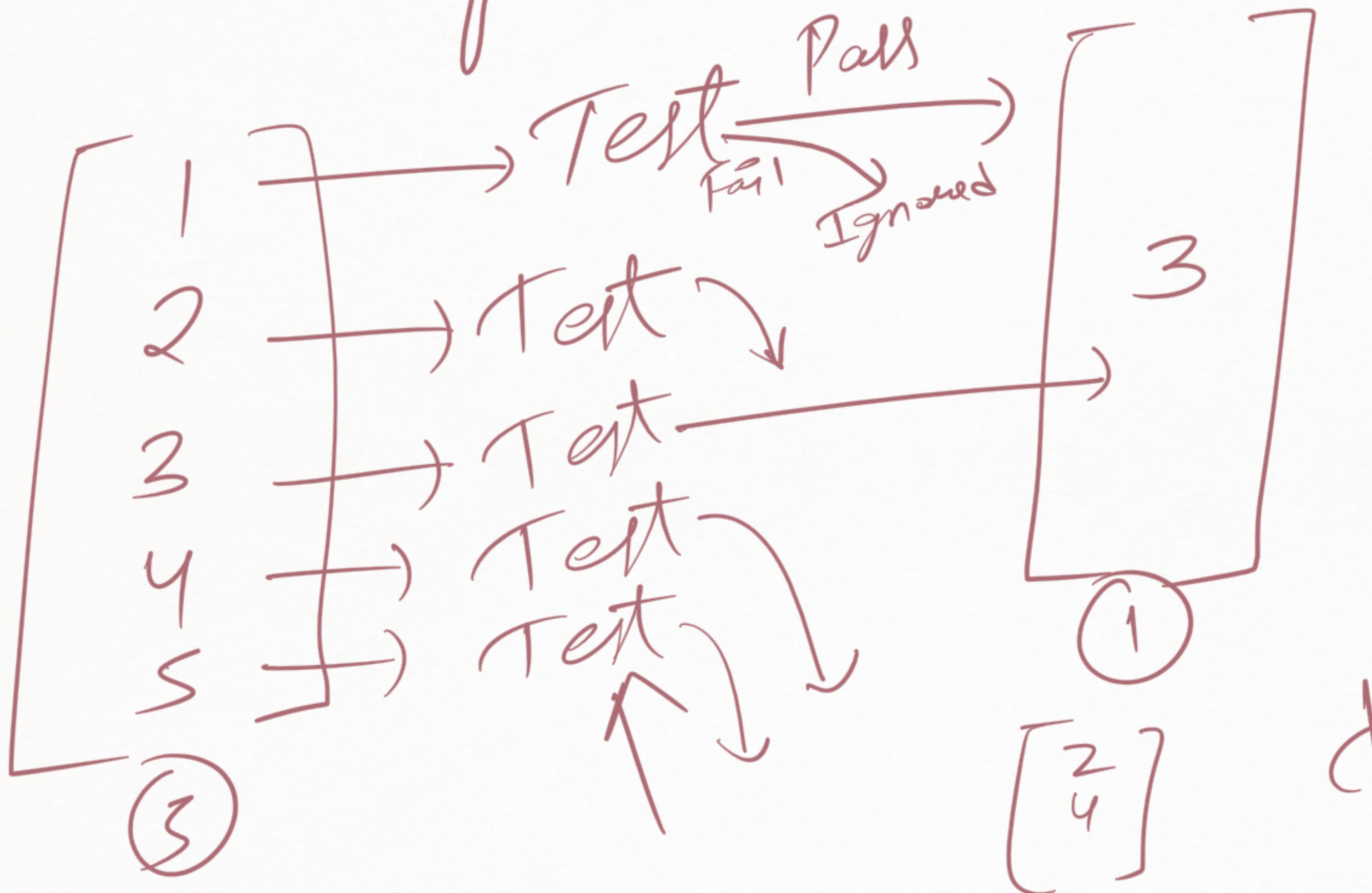


`def fun(x):  
 return x2`





filter  $\rightarrow$  final list kept in length  
than input



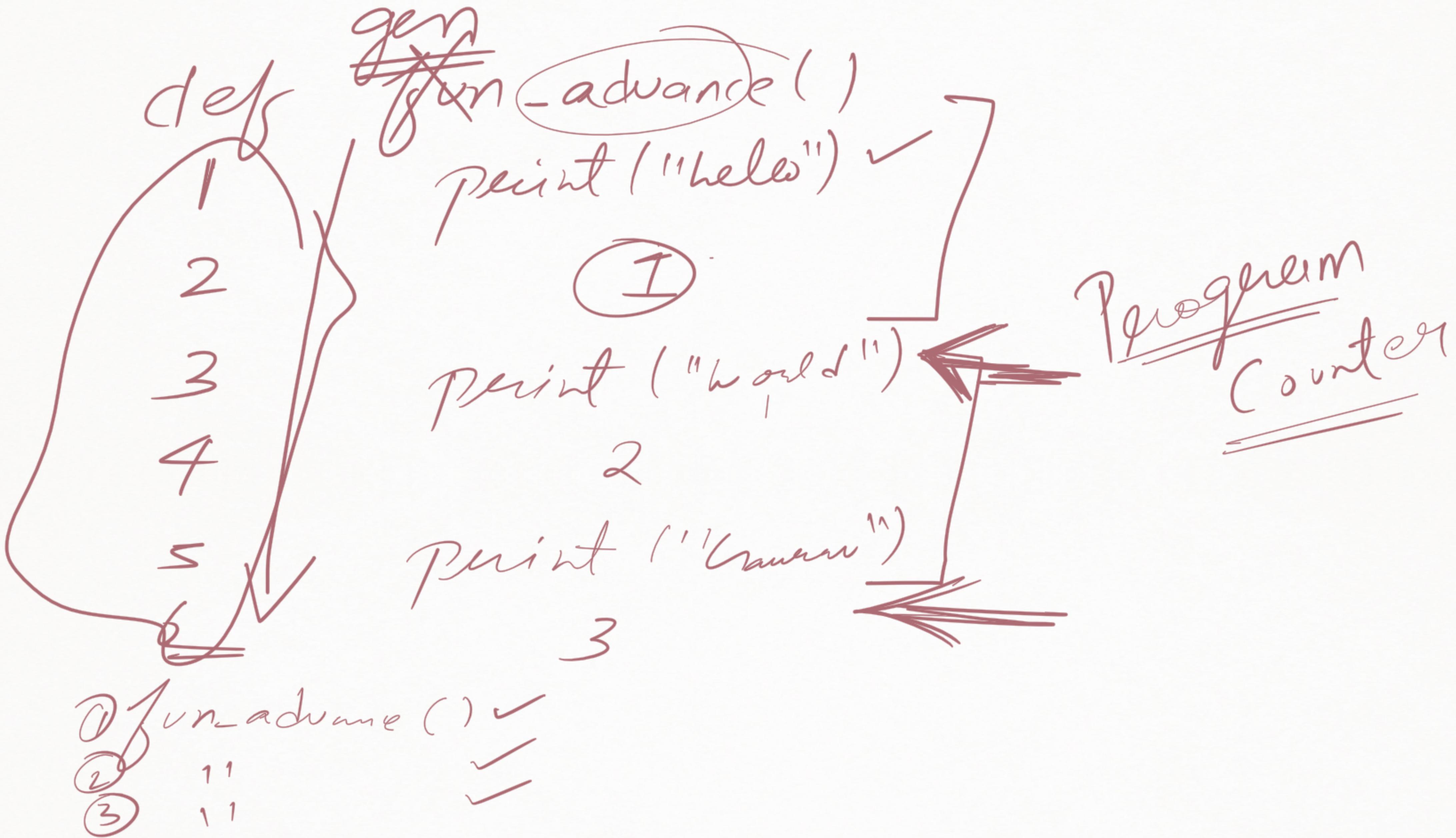
def Test(x):  
 if x == 3:  
 even true  
 else  
 even false

def even(x):  
 ✓

```
def fun_normal()
{
    -Print("hello") ✓
    return 1
    [ Print("world") ] X Never
    return 2
}
```

Print(fun\_normal()) ✓ => ? hello 1  
Print(fun\_normal()) => ? hello 1,

```
global x = 1
def fun_normal_2(x)
{
    if x == 1
        print("Hello")
        return 1
    else
        print("World")
        return 2
print(fun_normal_2(1)) =>? Hello!
print(fun_normal_2(2)) =>? World 2
```



# How to make a Gen function

```
def gen-fun():
    print("hello")
    yield ① ✓ P.C
    print("world")
    yield ② ←
```

\* Yield

gen-fun() → hello 1  
gen-fun() → world 2

# How to make iterator from it?

