#flo #inclass

---

1 | **Linear maps!**

when we hear linear, we should think: addition and scalar multiplication!

```
if it's in a box you should know it! -jana
```

homogenity: works nicely under SCAMUL - what does nicely mean? - addition and mapping is commutative? - can add, then send through map, which = send through map then add

- MULTI first then map = map then MULTI

this is called: **homomorphism**

```
title: homomorphism
structure-preserving map between two algebraic structures of the same type. -wiki
```

1.0.1 | **examples!**

KBxChapter3AReading#examples of linear maps

- how F^n to F^m works
  - inp: nx1 mat
  - oup: mx1 mat
  - which means: the org thing needs to be mxn mat
  - 3x1 -> 2x1

$ $\begin{bmatrix} 2 & -1 & 3 \\ 7 & 5 & -6 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 2x - y + 3z \\ 7x + 5y - 6z \end{bmatrix}$ $uhh..

- backwards shift!
  - [0 1 0 0 0]
  - [0 0 1 0 0]
  - this just drops the first element, and is essentially a repeat of above example of mapping to a lower dimension
  - #question wait how do u actully define this operation? just create an example with arbitrary elems?

if you know where the basis goes, then you know where everything else goes! @3b1b talked about this KBxChapter3AReading#linear maps and basis of domain

1.0.2 | algebraic operations

this is how we combine maps!

- how we are used to adding functions

we get.. addition and SCAMUL which makes:

`[[file:KBxL(V,W).org][KBxL(V,W)]] a [[file:KBe20math530refVectorSpace.org][KBe20math530refVectorSpace]]`

`inp: nx1`, `oup: mx1` for KBxChapter3AReading#product of linear maps
if the domain of one matches the KBxCodomain of another