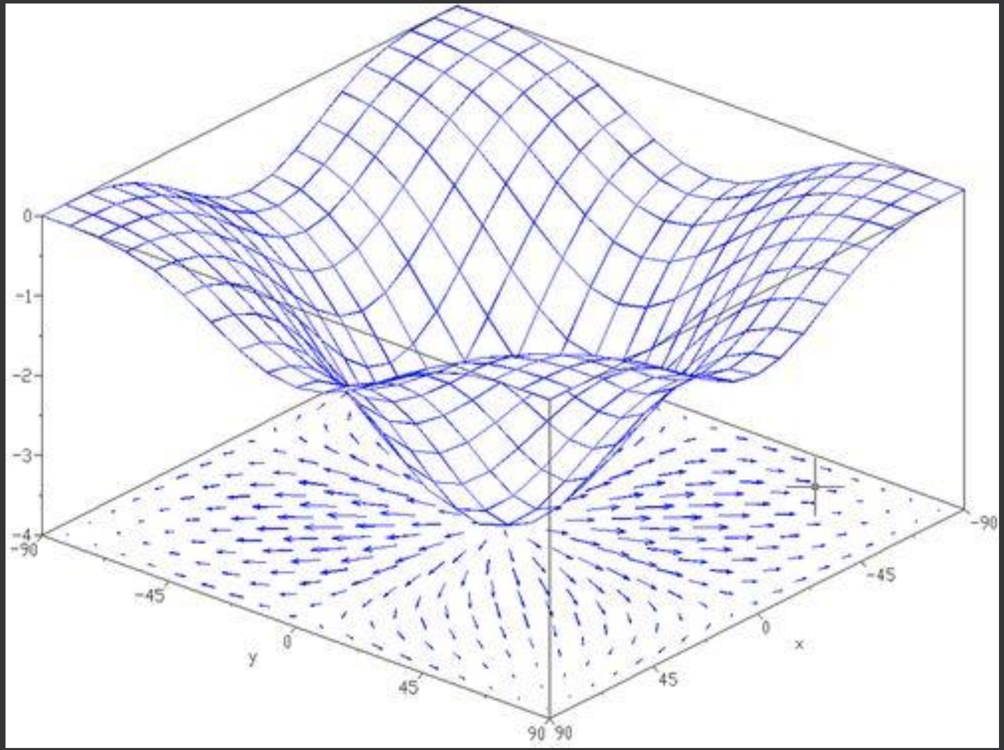


Vector Operations

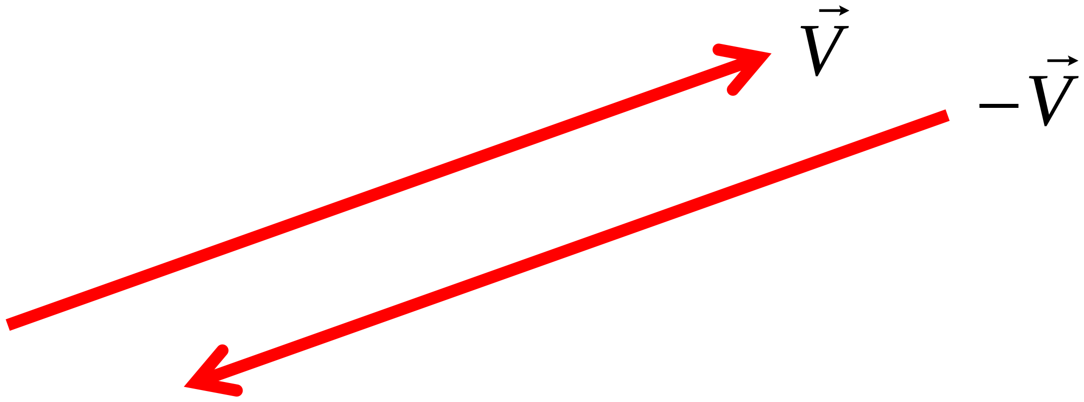
Lecture 2

Vector properties

- We’ll look at a number of Vector Operations
- Ways of looking at each:
  - **Symbolically**: With a symbol.
  - **Numerically**:
    - The steps to “do” this operation
    - The way you’ll implement it in Python
  - **Graphically**: With a picture
    - Usually the most complicated / useful
    - Often many interpretations (watch for the point / vector distinction)

Vector Negation (2.5)

- **Symoblically**: $-\vec{v}$
- **Numerically**:
  - Negate all of the components to produce a *new* vector
  - Define the __neg__ method in Python:
    - Should return a new VectorN.
- **Graphically**:



Vector-scalar multiplication (2.6)

- **Symbolically** (all of these are requivalent):
  - $\boldsymbol{k\vec{v}}$
  - $\boldsymbol{\vec{v}k}$ (quiz: what's the term connecting this and the previous?)
  - $\boldsymbol{k * \vec{v}}$
    - (try not to use the ● or 'x' symbol sometimes used in scalar multiplication – it means something different with vectors)
  - $\boldsymbol{\vec{v} * k}$

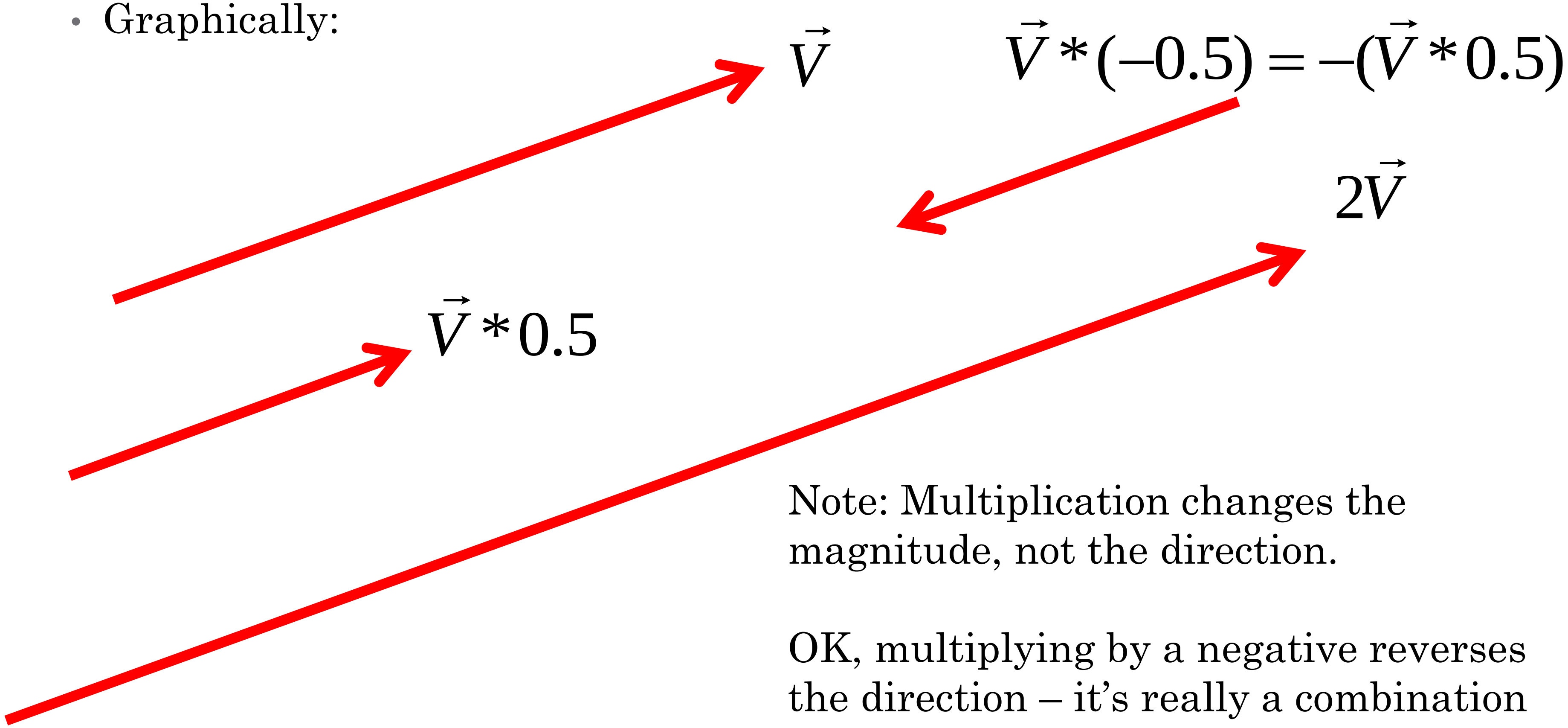
- **Numerically**:
  - Just multiply the components by the scalar to get a new vector.
  - E.g.:

$$\begin{aligned} &[4 \quad 7 \quad -2] * 3 \\ &= [4 * 3 \quad 7 * 3 \quad -2 * 3] \\ &= [12 \quad 21 \quad -6] \end{aligned}$$

  - In python, define the __mul__ method and the __rmul__ method (why rmul?)

Vector-scalar multiplication, cont.

- Graphically:



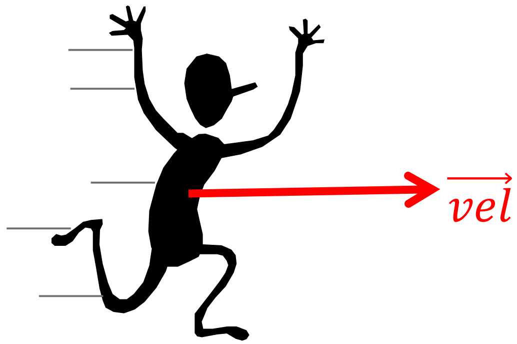
Note: Multiplication changes the magnitude, not the direction.

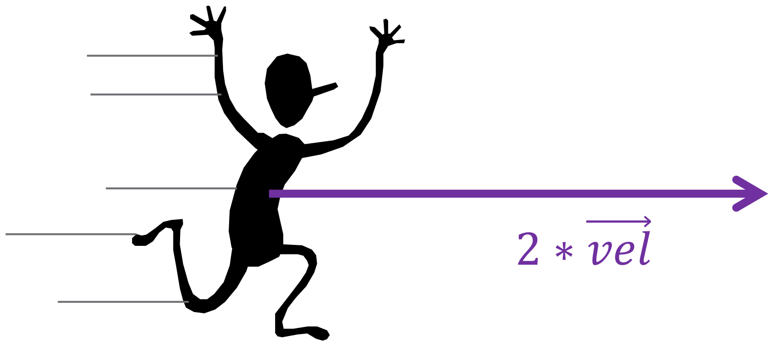
OK, multiplying by a negative reverses the direction – it's really a combination multiplication & negation

Note2: Point multiplication doesn't really have a (useful) application.

An Example

- Suppose you're travelling along the ground to the East (where East is +z) at 10 mph: (0,0,10)
- You double your speed: Now your velocity is (0,0,20)





Vector-scalar division

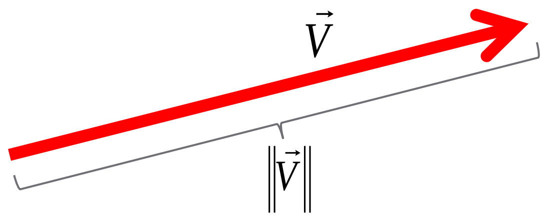
- **Symbolically**: $\vec{v}/k$
  - Note: $k/\vec{v}$ isn't defined
- **Numerically**:
  - $\frac{\vec{v}}{k} = \vec{v} * \frac{1}{k}$
    - Which is just vector-scalar multiplication
  - In python (3.x+), define the __truediv__ method.
  - There is a __rtruediv__ method we'll define, but we'll always raise an exception (why?)
- **Graphically**: Not much difference between this and vector*scalar…

Vector Length (2.8)

- I'll usually called it **magnitude**
  - So there's no confusion with the totally-unrelated len function (which tells us the dimension)
- **Symbolically**: $\|\vec{v}\|$
  - NOT absolute value
  - Magnitude is always a scalar value (regardless of dimension)
- **Numerically**:
  - The distance from the vector's tail to its head.
  - How do we do it in 2D?
  - Pythagorean Theroem!
  - It's the same form in any dimension.

$$\|\vec{v}\| = \sqrt{{v_1}^2 + {v_2}^2 + {v_3}^2 \ldots + {v_n}^2}$$

  - In python, define a **magnitude** method
    - Because there's no "hook" for this symbol…

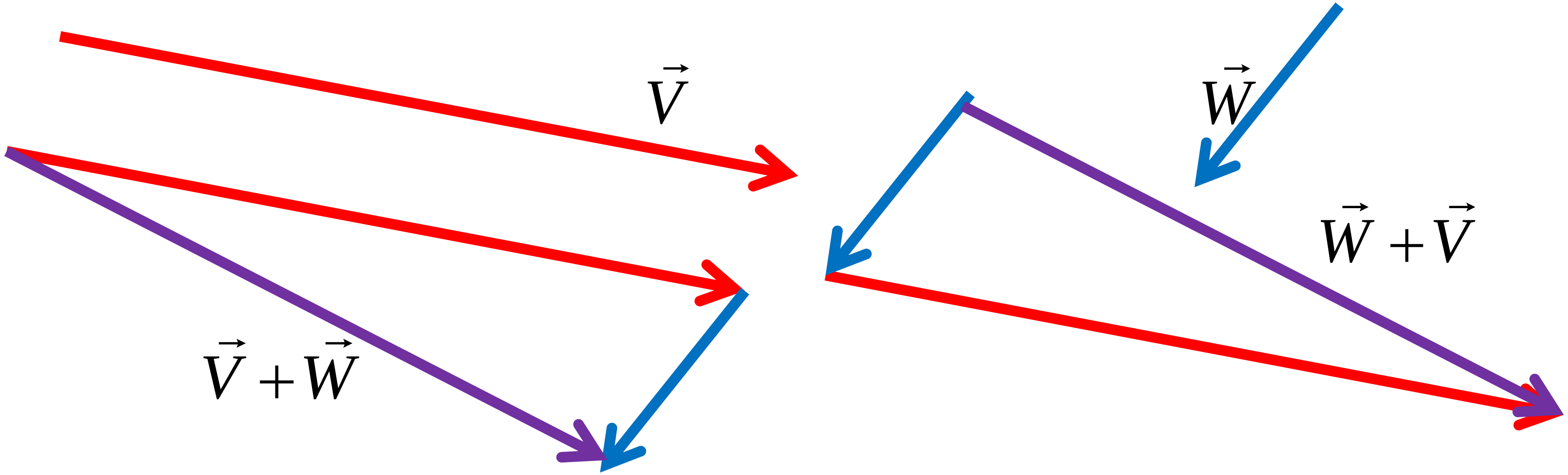


Vector addition & subtraction (2.7)

- **Symbolically:**
  - $\vec{\boldsymbol{v}} - \vec{\boldsymbol{w}}$
  - $\vec{v} + \vec{w}$
- **Numerically**:
  - Addition: Just add the components to get a new vector (order is not important)
    - The __add__ method in python
  - Subtraction: Subtract the components to get a new vector (order **IS** important)
    - The __sub__ method in python
    - Note: $\vec{v} - \vec{w} = \vec{v} + (-\vec{w})$
      - This is a negation followed by an addition
      - Useful in our graphical interpretation

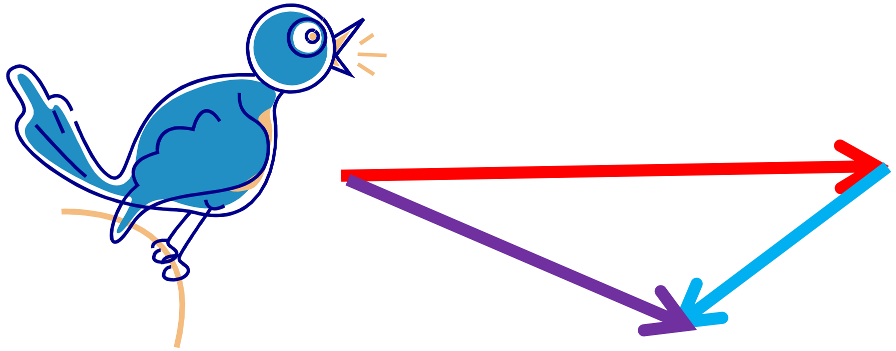
Vector Addition, cont.

- **Graphically:**
  - Aka the "Parallelogram Method"
  - Align the second tail on the first head
  - The result is a new vector from the first tail to the last head
  - Since addition is commutative, we can reverse the order



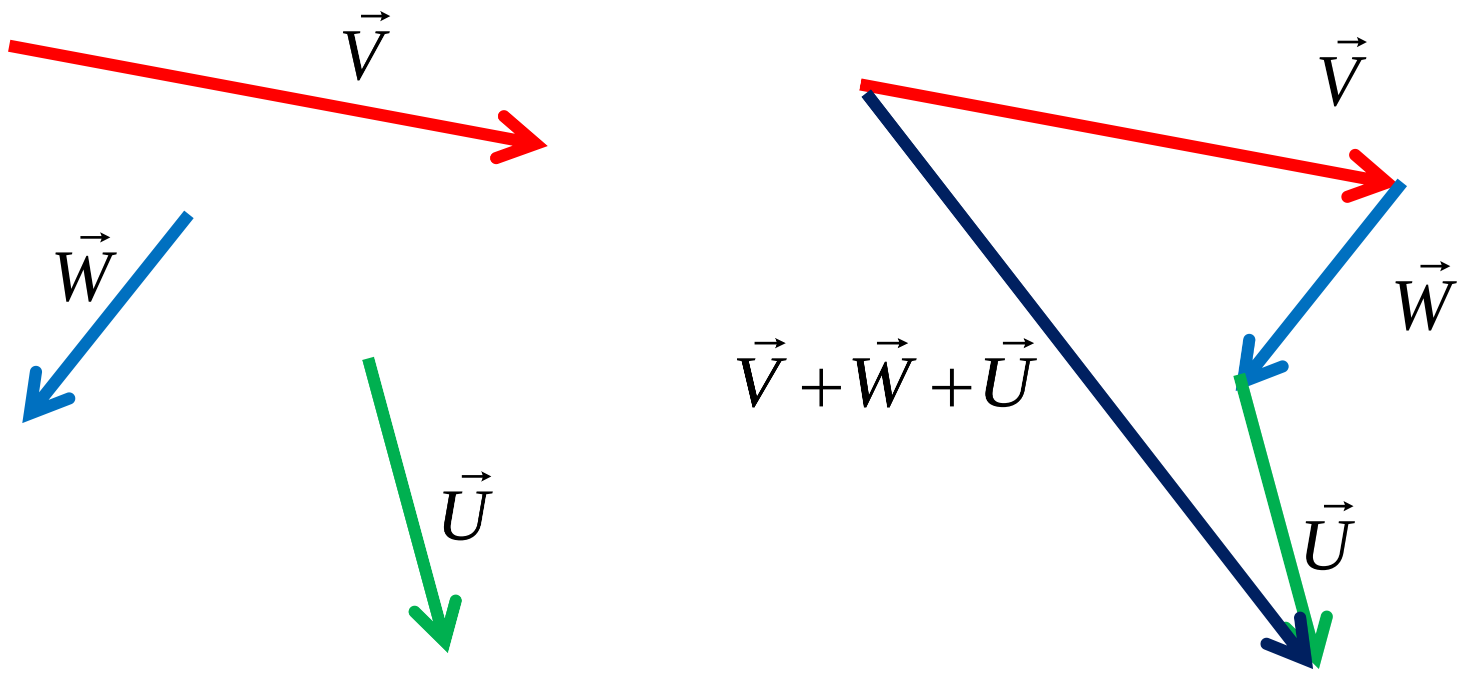
Vector Addition, Example #2

- A bird flying with a velocity of $[20 \quad 0]$
- A wind blows with a velocity of $[-9 \quad -8]$
- The net velocity of the bird is $[20 \quad 0] + [-9 \quad -8] = [11 \quad -8]$



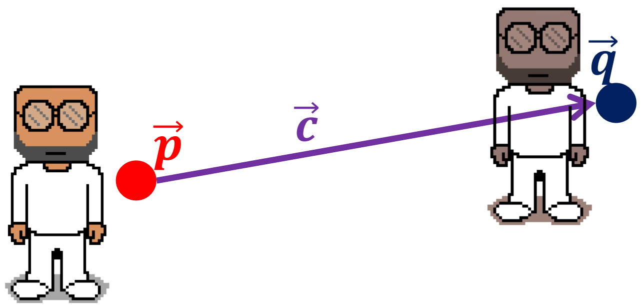
More on Vector Addition

- You can add more than one Vector:
  - Just line them up tail-to-head.
  - The net result goes from the first tail to the last head.
  - Still commutative.



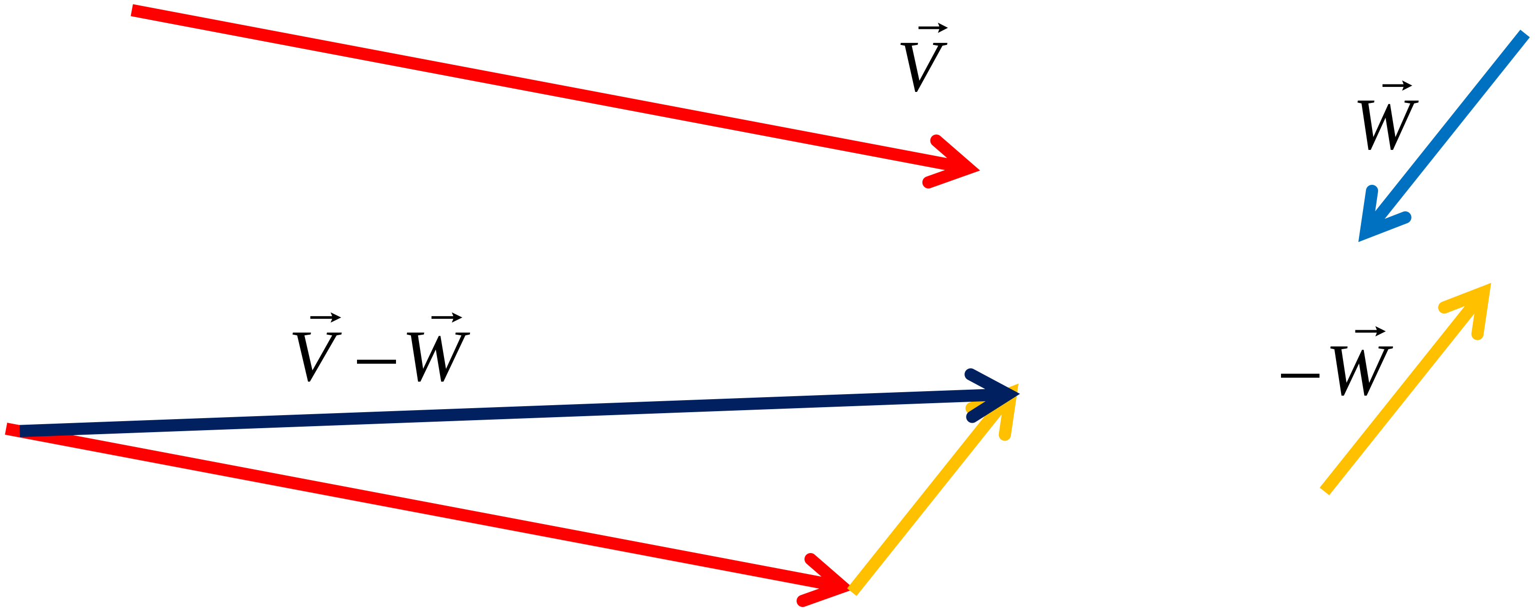
Point Math

- Point addition is *meaningless*
  - Geometrically, you can't / won't add two points
- Point + Vector = Move something in a given direction
  - Very useful!
  - Ex:
    - You are floating in the ocean at position $\vec{p}$
    - A strong current with direction $\vec{c}$ pushes you
    - Your final location is $\vec{q} = \vec{p} + \vec{c}$



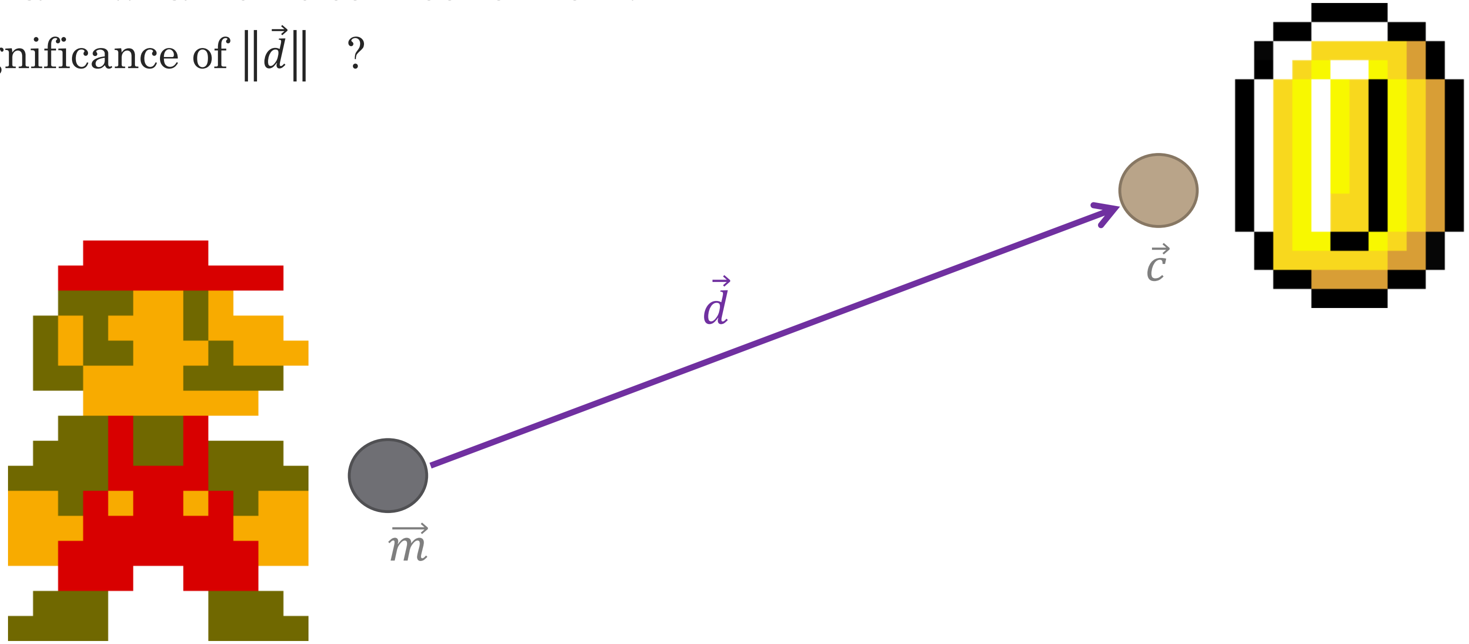
Vector Subtraction

- Remember, V – W is the same as V + (-W)
- So…geometrically:



Point Subtraction

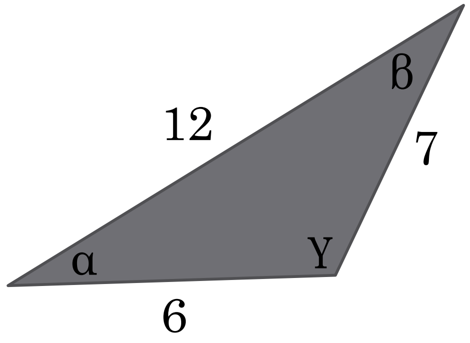
- A – B: Produces the vector *direction* to go from B to A
  - Remember: “destination” – “start”
- E.g.
  - $\vec{d} = \vec{c} - \vec{m}$
  - What is $\vec{m} - \vec{c}$ and what is its connection to $\vec{d}$?
  - What's the significance of $\|\vec{d}\|$?



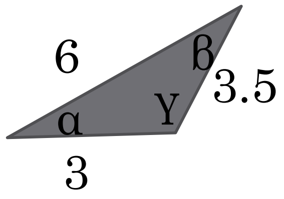
Zero Vector (2.3.2)

- A property of a vector, not an operation
- **Symbolically**:
  - $\vec{\emptyset}$
  - **Or** $\vec{v} = [\mathbf{0} \quad \mathbf{0} \quad \mathbf{0}]$
- **Numerically**: A vector is the zero vector if it has 0's for all its components.
- **Graphically** (hard to draw a picture…):
  - A zero-length **vector** (i.e. no displacement)
  - A **point** at the origin.
- Interesting fact: $\vec{0} + \vec{a} = \vec{a}$
  - Vector additive identity
  - This is another (better) way of making the point – vector connection
    - We're adding the **vector** $\vec{a}$ to the point 0, getting the **point** $\vec{a}$

Vector normalization(2.9)

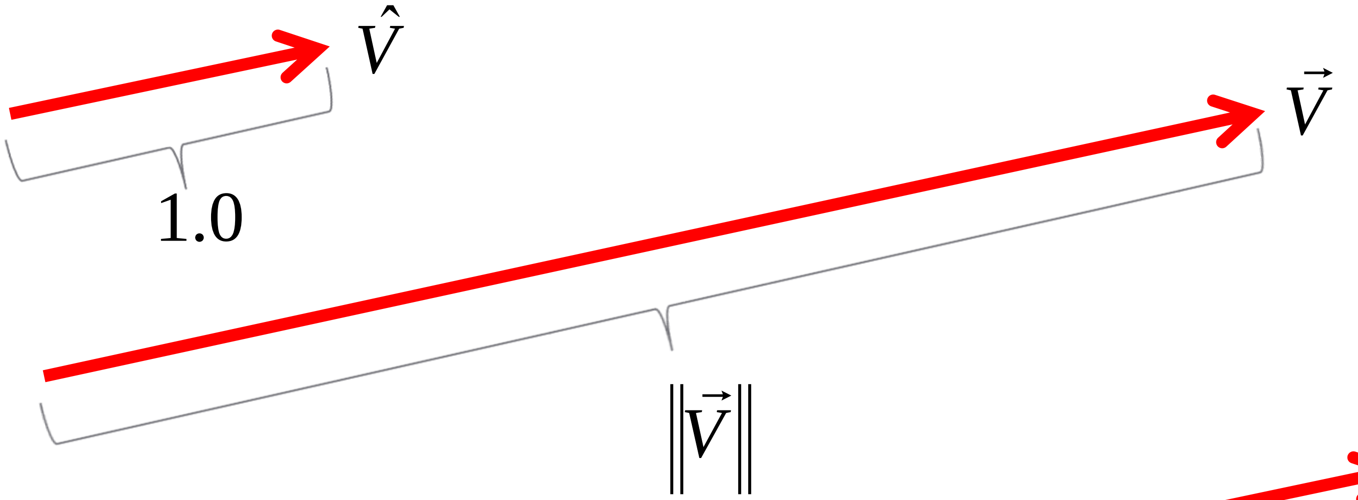


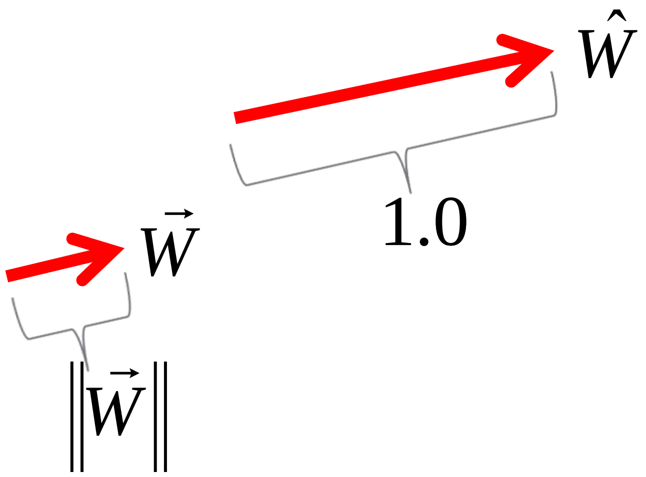
- A vector is **unit length** if it has a magnitude of exactly 1.0.
- Indicated symbolically by: $\hat{\boldsymbol{v}}$
  - Still a vector, just a *special* type of vector.
  - Nice property: $||k\hat{v}|| = k$
- **Normalization** is the process of "shrinking" or "growing" an arbitrary vector so that:
  - It's direction is unchanged.
  - It's length becomes 1.0
  - **Law of Similar Triangles**: *"Two triangles are similar iff the ratio of two sides is the same in both triangles (equivalently: multiply all side-lengths by a constant to produce a similar triangle)"*



$$\hat{\boldsymbol{v}} = \frac{\vec{\boldsymbol{v}}}{||\vec{\boldsymbol{v}}||}$$

Vector Normalization, cont.





$$\vec{v} = [2 \quad 1]$$

$$\|\vec{v}\| = \sqrt{1^2 + 2^2} \approx 2.236$$

$$\hat{v} = \frac{[2 \quad 1]}{\|\vec{v}\|} = \left[\frac{2}{2.236} \quad \frac{1}{2.236}\right]$$

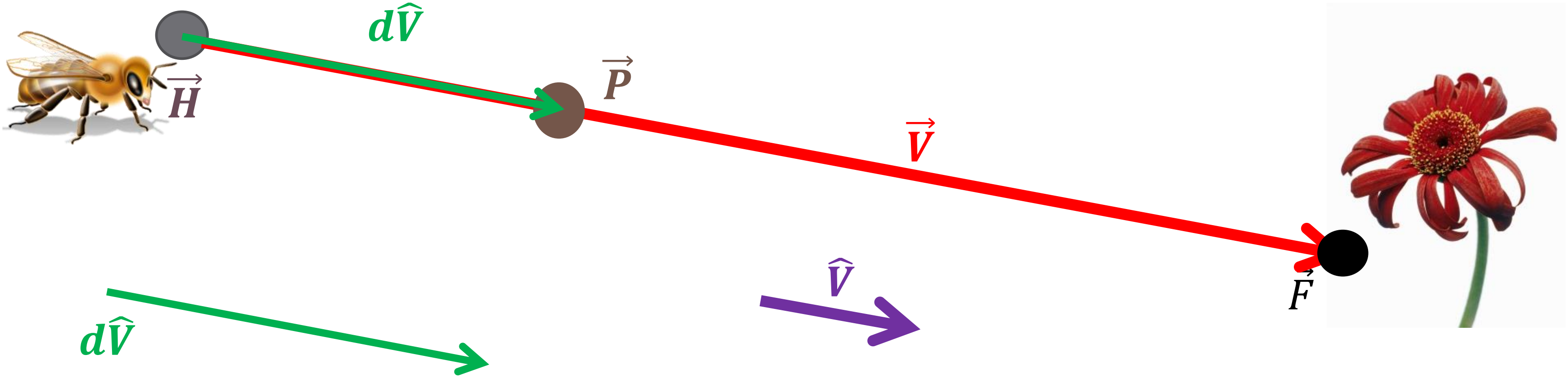
$$\approx [0.8994 \quad 0.4472]$$

$$\|\hat{v}\| \approx \sqrt{0.8944^2 + 0.4472^2} = 0.9997$$

Quiz-like problem

$$\boldsymbol{Let\ \vec{V} = \vec{F} - \vec{H}}$$
$$\boldsymbol{\hat{V} = \frac{\vec{V}}{\|\vec{V}\|}}$$
$$\boldsymbol{\vec{P} = d\hat{V} + \vec{H}}$$

- Given:
  - $\vec{H}$: Honey Bee's position
  - $\vec{F}$: Flower's position
  - d: the distance the honeybee can travel in 1 minute
- Find (symbolically) $\vec{P}$, the honey bee's position after 1 minute.
  - You can assume the bee won't be able to reach the flower in 1 minute.



Normalization example, asked Numerically

- Why no numbers?
  - The symbolic solution (algorithm) is *much* more valuable than a number answer.
  - Why? Just substitute *any* number in for the placeholders and get a value!

Symbolic solution

Let $\vec{V} = \vec{F} - \vec{H}$

$$\hat{V} = \frac{\vec{V}}{\|\vec{V}\|}$$

$$\vec{P} = d\hat{V} + \vec{H}$$

- Example:
  - $\vec{H} = [500 \quad 300]$: Honey Bee's position
  - $\vec{F} = [700 \quad 400]$: Flower's position
  - d = 50: the distance the honeybee can travel in 1 minute
  - Now just plug-and-go

$$\vec{V} = \vec{F} - \vec{H}$$
$$= [700 - 500 \quad 400 - 300]$$
$$= [200 \quad 100]$$

$$\|\vec{V}\| = \sqrt{200^2 + 100^2} \approx 223.6$$

$$\hat{V} = \frac{\vec{V}}{\|\vec{V}\|} = \left[\frac{200}{223.6} \quad \frac{100}{223.6}\right]$$
$$= [0.894 \quad 0.447]$$

$$\vec{P} = d\hat{V} + \vec{H} = 50[0.894 \quad 0.447] + [500 \quad 300]$$
$$= [544.7 \quad 322.35]$$