Geometry and Geometric Programming

CMSC425.01 Fall 2019

Administrivia

- Questions on Project 1
  - Note where the material is on the web site
  - Handouts and Project page
- Starting on theory track in course
  - Homeworks coming
  - Class periods will include problems, live work on document cameras
  - Multiple solutions to most problems – will accept most, should know best

Today’s question

What’s the point?
Points, vectors, lines, rays and other geometric objects

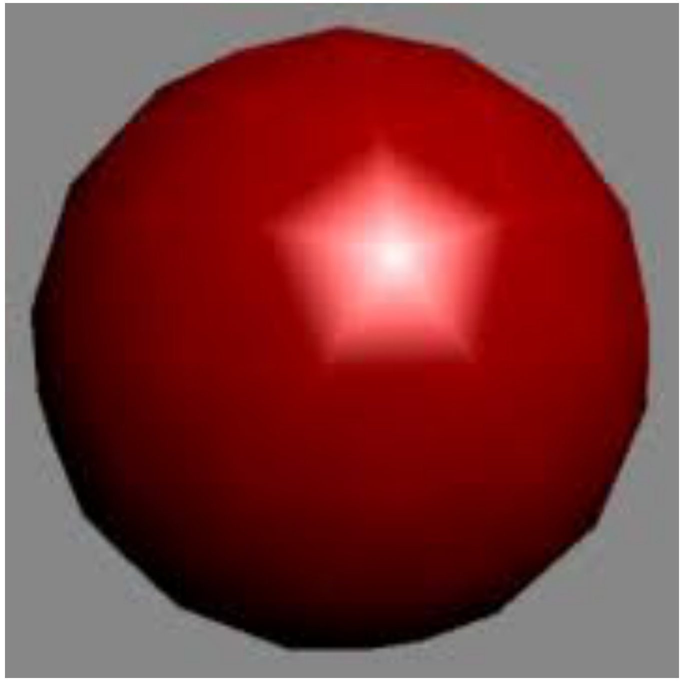
What math is needed for games?

What math is needed for games?

- ***Unlimited***
  - Games are simulations of anything you want, so any math
  - Primarily general motion, physics, light
  - But also
    - Flight sim – aerodynamics, fluid dynamics
    - Accurate space game – astrodynamics
    - SimCity – social and physical mechanisms, networks
  - Math 240 (linear), physics, Math431 (math for graphics)

What math is needed for games?

- ***But we can cheat***
  - Simulations don't have to be accurate
  - Only look and feel right
  - *Or,* lean in to artificiality …

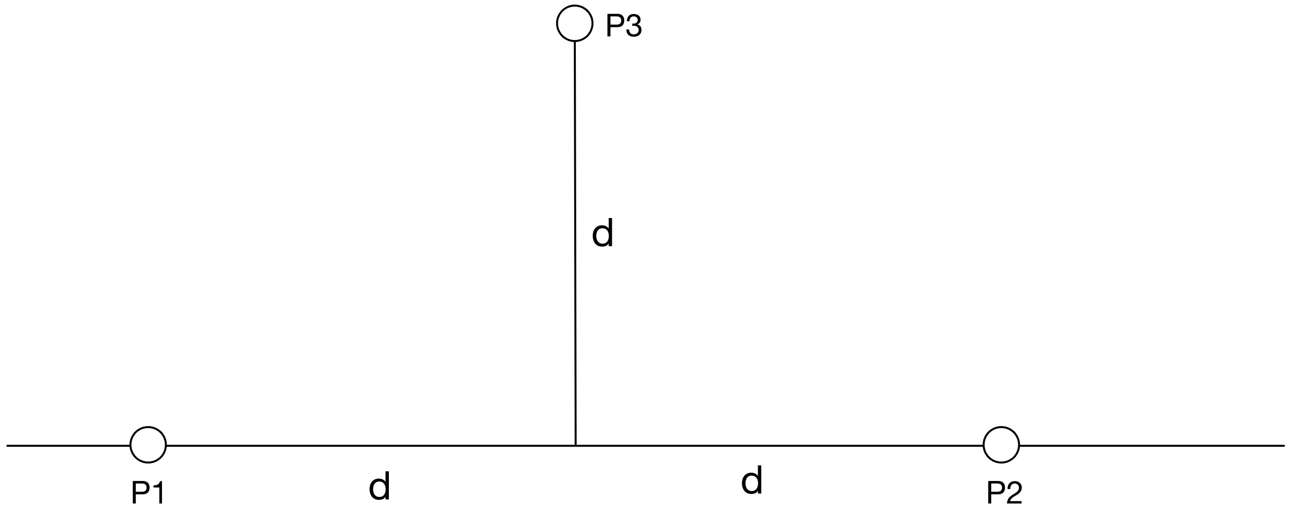


Gouraud shading – approximates smooth lighting on underlying mesh

Sajidur Rahman

Problem 1: What is the point - really

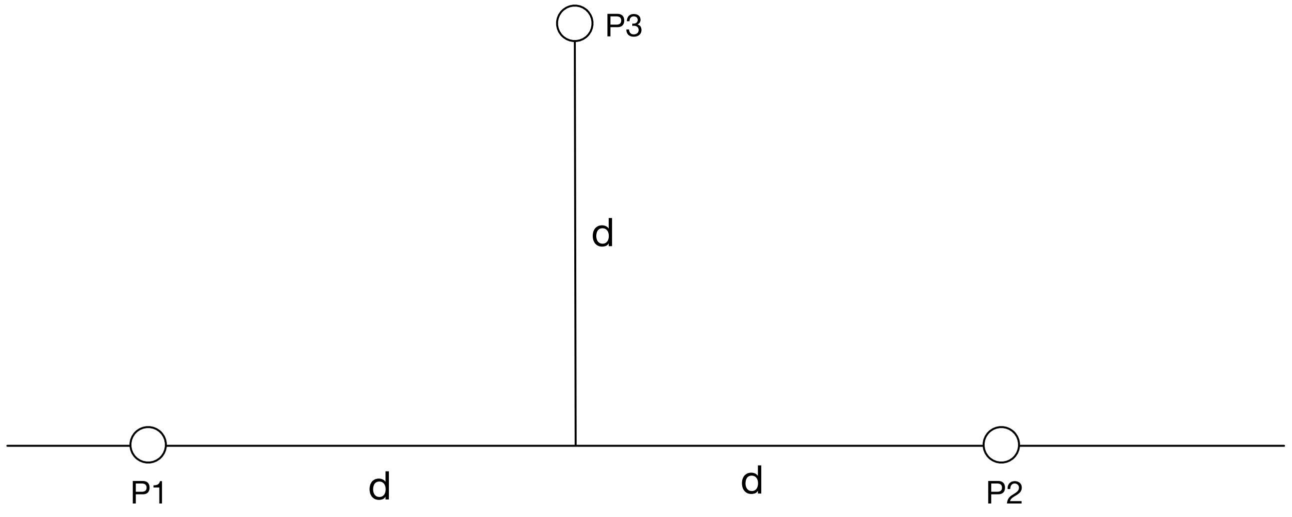
- Given P1, P2 on the x-axis
- What is P3?
  - A distance d above the midpoint



Problem 1: What is the point - really

- Given P1, P2 on the x-axis
- What is P3?
  - A distance d above the midpoint

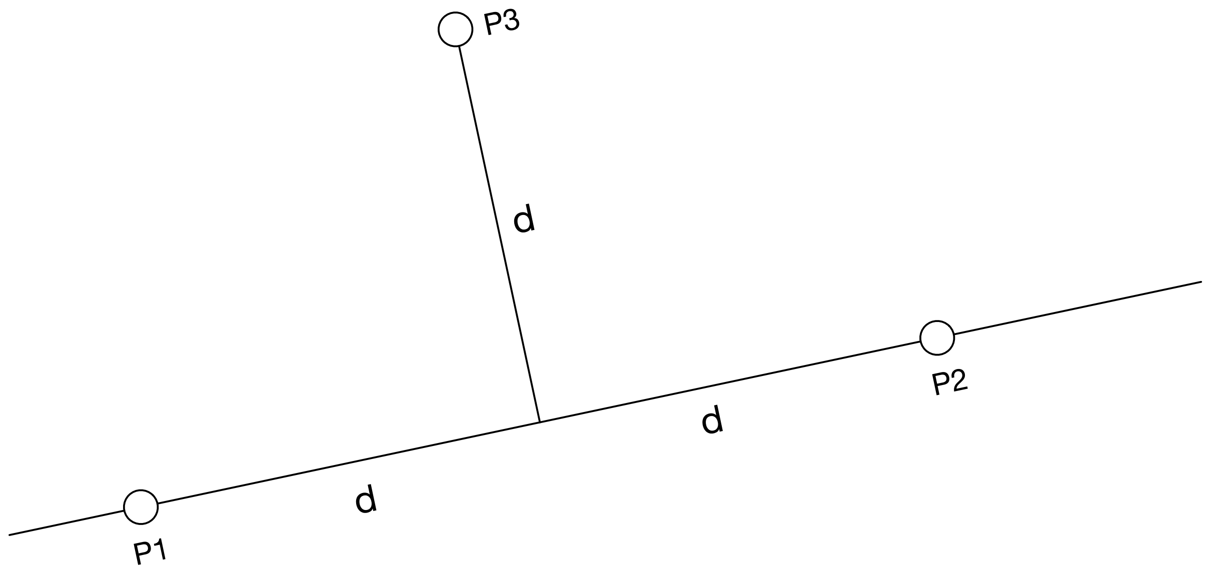
- Solution 1:
- P3.x = P1.x + (P2.x-P1.x)/2
- P3.y = (P2.x-P1.x)/2



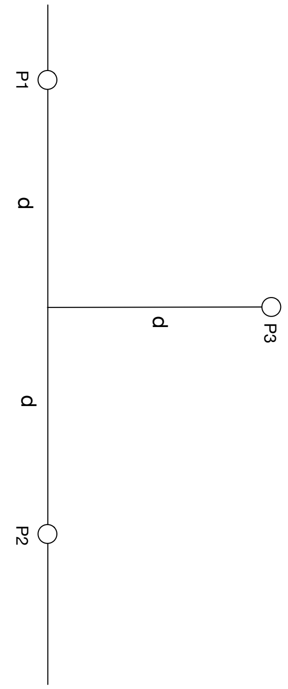
Problem 2: What is the point, too

- Given P1, P2 on any line
- What is P3?

- Midpoint displacement …



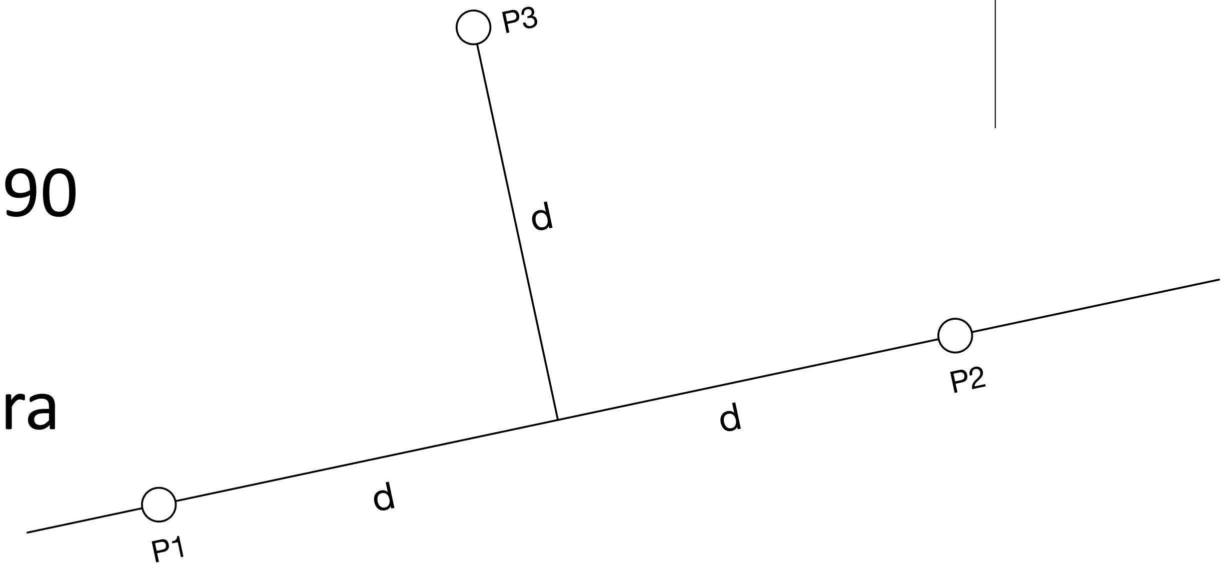
Problem 2: What is the point, too



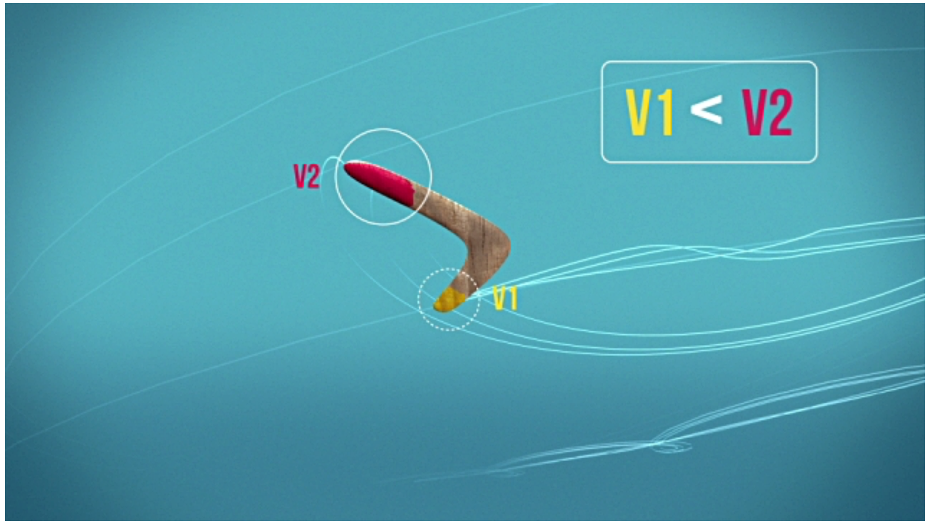
- Given P1, P2 on any line
- What is P3?

- Need general solution
- Works for any angle including 90

- Will review and develop algebra of points, lines, vectors, rays, and related shapes



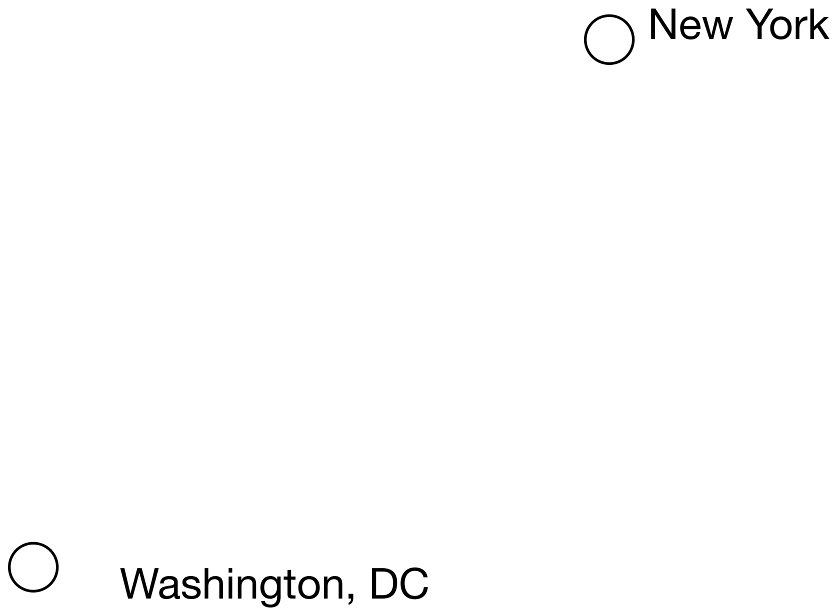
Other problems



- Geometric constructions – create shapes
  - Midpoint displacement mountains
- Transformations – move or position objects
  - Twirling boomerang: rotate and move?
- Orientation – which way should we point or move?
  - How rotate airplane to attack or escape threat?
- Light – how do light rays reflect/refract off objects?
  - Transparent bowl – how balance shiny and transparent?

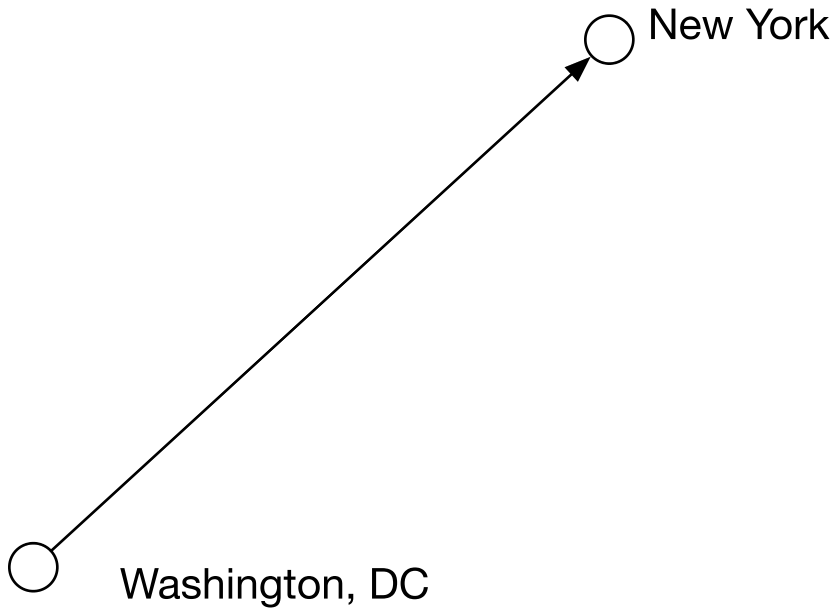
Problem 3: What does DC + NY mean?

- What does adding two locations mean?



Problem 3: What does DC + NY mean?

- What does adding two locations mean?
- What makes sense:
  - Distance | DC-NY |
  - Vector DC-NY
  - Orientation – angle of direction
- What doesn't:
  - DC+NY Where is that?



Points vs. Vectors as different types

- Point (x, y)
  - 2 or 3D
  - Location – place

  - Makes sense to subtract
  - Does *not* make sense (always) to add, multiply by scalar (scale), take dot product

  Problem: Vector3 used for both pts and vectors in Unity (and elsewhere)

  Moral: pay attention to p vs. v

- Vector $< x, y >$
  - 2 or 3D
  - Displacement

  - Makes sense to add, subtract, multiply by scalar (scale), take dot product

  - Conversion: v = p – p
    or p = p + v

Affine geometry

- Scalars α, β, γ (or a, b, c)
- Points p, q, r
- Vectors (free) $\vec{u}, \vec{v}, \vec{w}$ (zero vector $\vec{0}$ with $\vec{u} = \vec{u} + \vec{0}$)
- Operations

scalar-vector multiplication	$v \leftarrow s \times v$, $s \,/\, v$
scalar-vector addition	$v \leftarrow v + v$, $v - v$
point-point different	$v \leftarrow p - p$
point-vector addition	$p \leftarrow p + v$, p$-v$

Affine operations

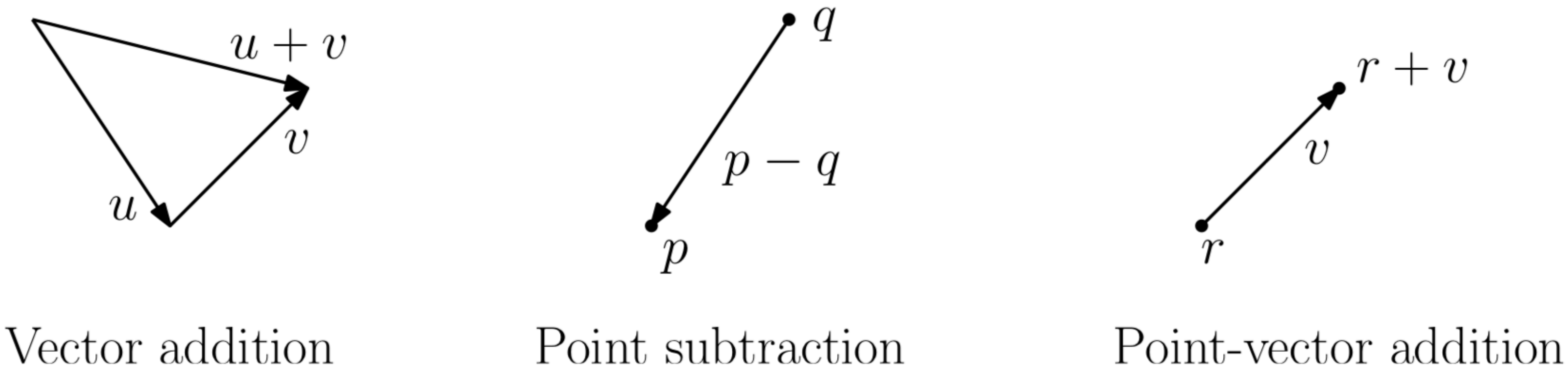
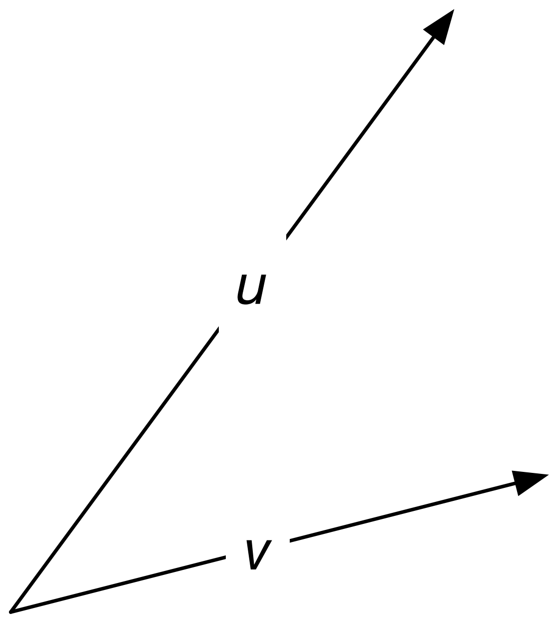


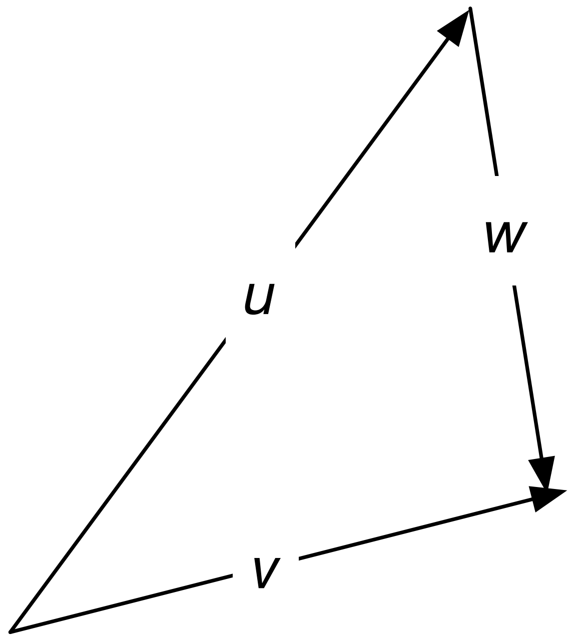
Fig. 1: Affine operations.

Questions

Is vector addition commutative?

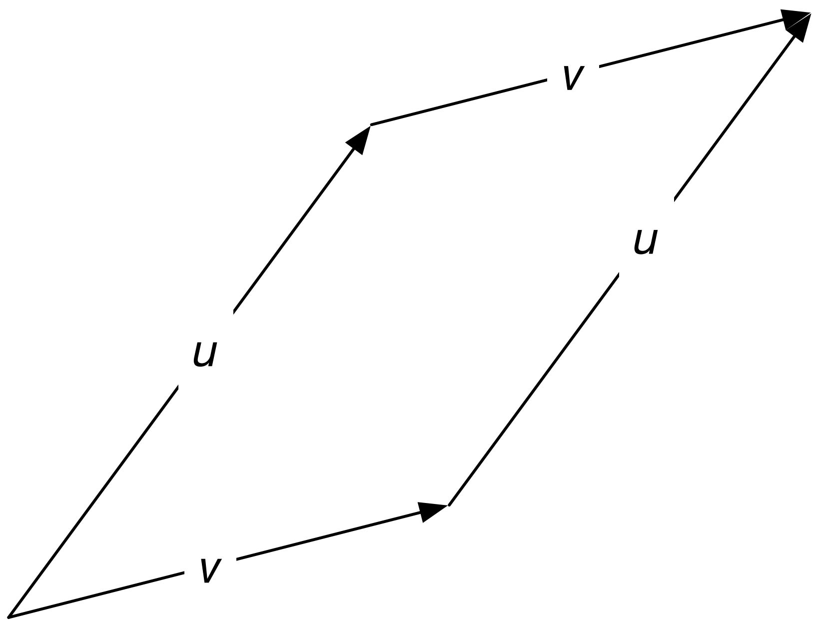


What's the vector *w* in this diagram?

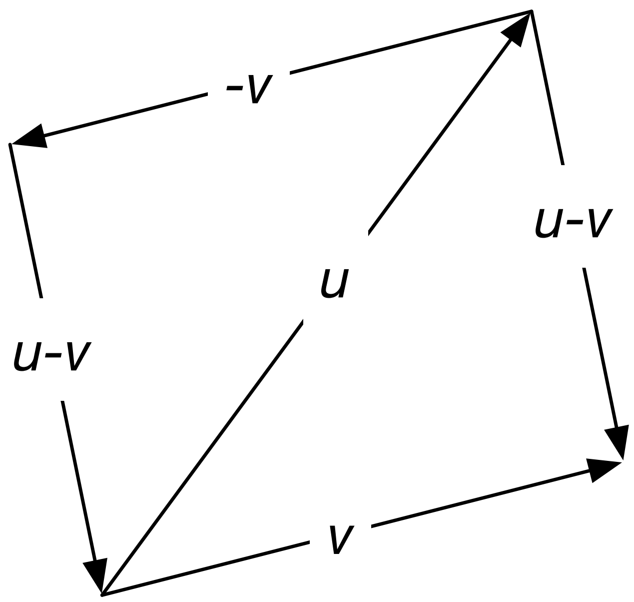


Questions

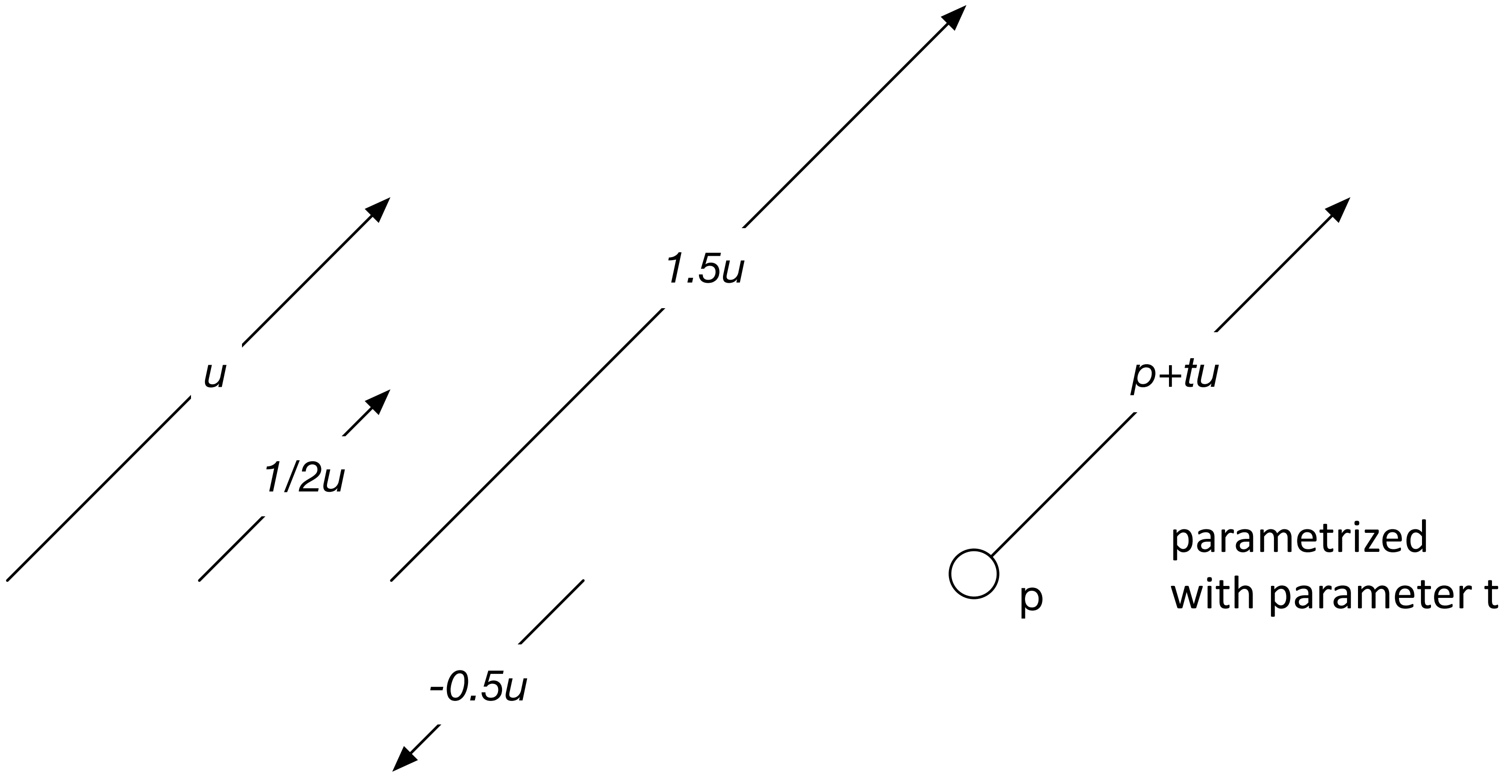
Is vector addition commutative?



What's the vector *w* in this diagram?

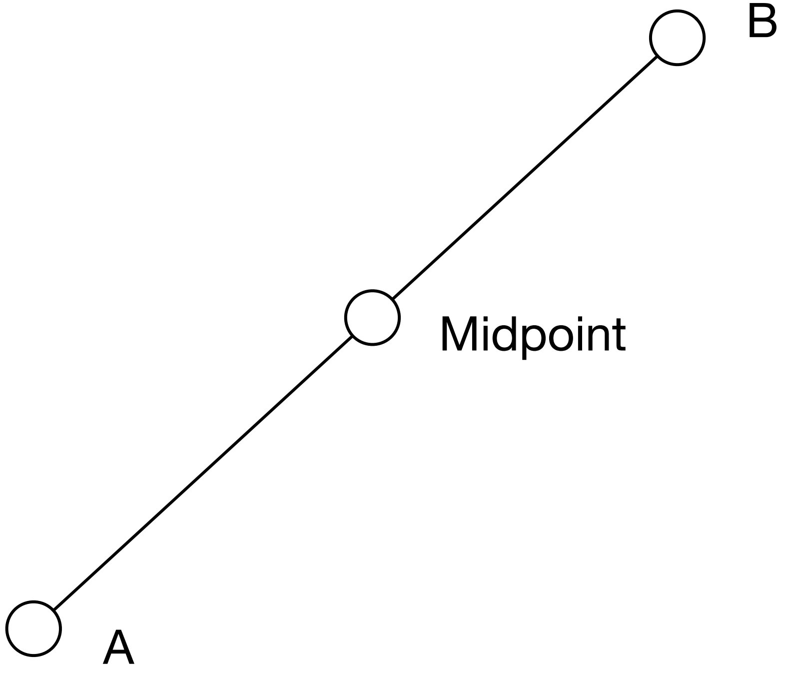


Vector scaling



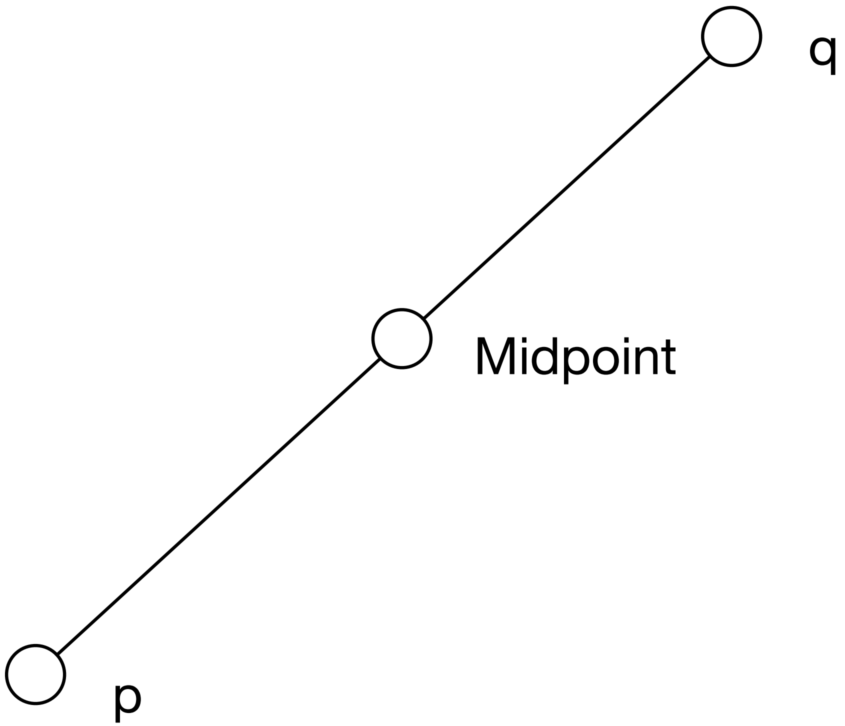
Problem 4: Affine combinations

- Midpoint between two points?



Problem 4: Affine combinations

- Midpoint between two points?
- $m = \frac{p+q}{2}$
- $m = (\frac{px+qy}{2}, \frac{py+qy}{2})$

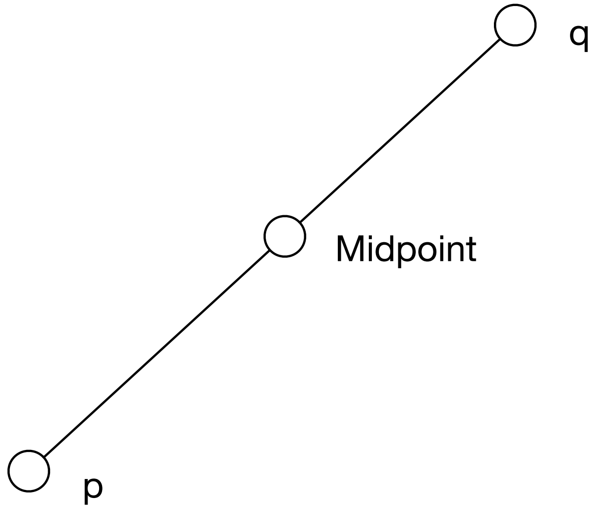


Problem 4: Affine combinations

- Midpoint between two points?

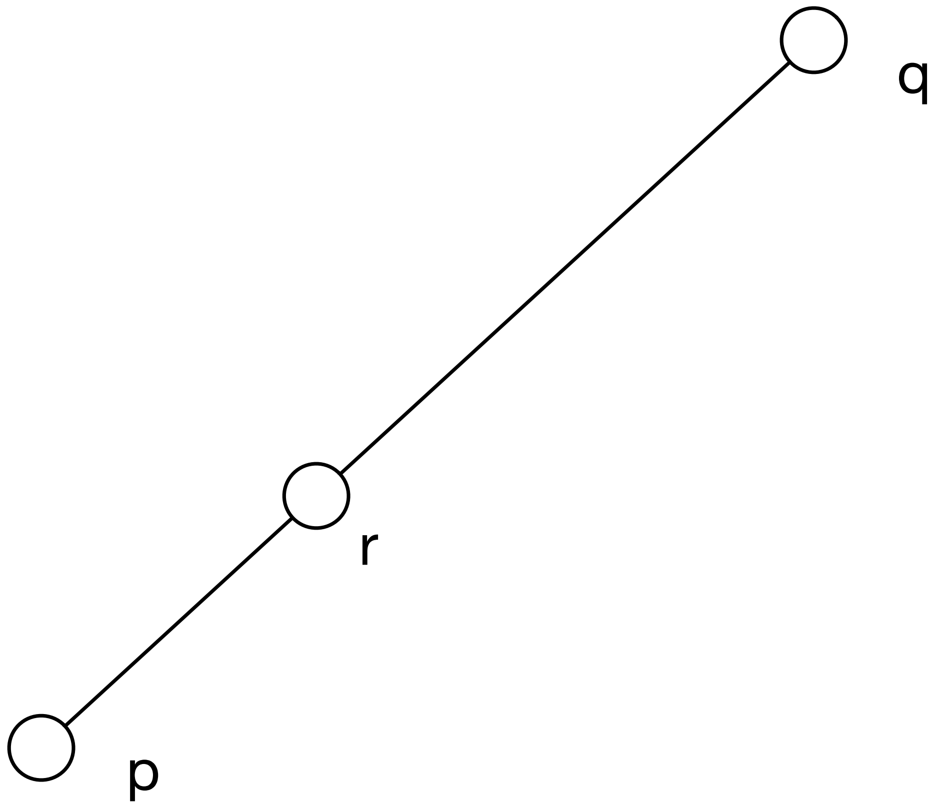
- $m = \frac{p+q}{2}$ ***coordinate free***

- $m = (\frac{px+qy}{2}, \frac{py+qy}{2})$ ***coordinate based***



- Coordinate free formulas are better – generalize to 3D

Problem 6: Line between two points?



Line between two points?

- Version 1: coordinate based

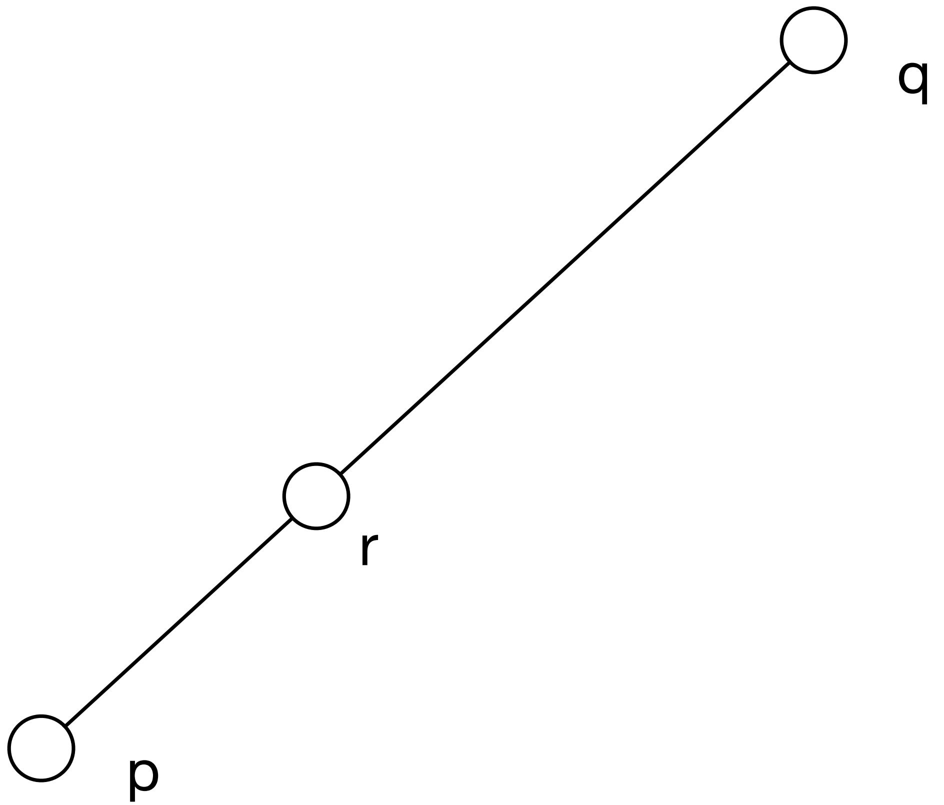
$$y = mx + y0$$
$$m = dy/dx$$

- Version 2: vector + coordinate free

$$r = p + \alpha(q - p)$$

or

$$r = p + \alpha v$$



Problem 6: Line between two points?

- Version 1: coordinate based

$$y = mx + y0$$
$$m = dy/dx$$

- Version 2: vector + coordinate free

$$r = p + \alpha(q - p)$$
$$or$$
$$r = p + \alpha v$$

- How handle vertical line?
  - Version 1: dx is zero, need x=f(y)
  - Version 2: v = < 1,0 >

Problem 7: Using vector line equation

$$r = p + \alpha v$$

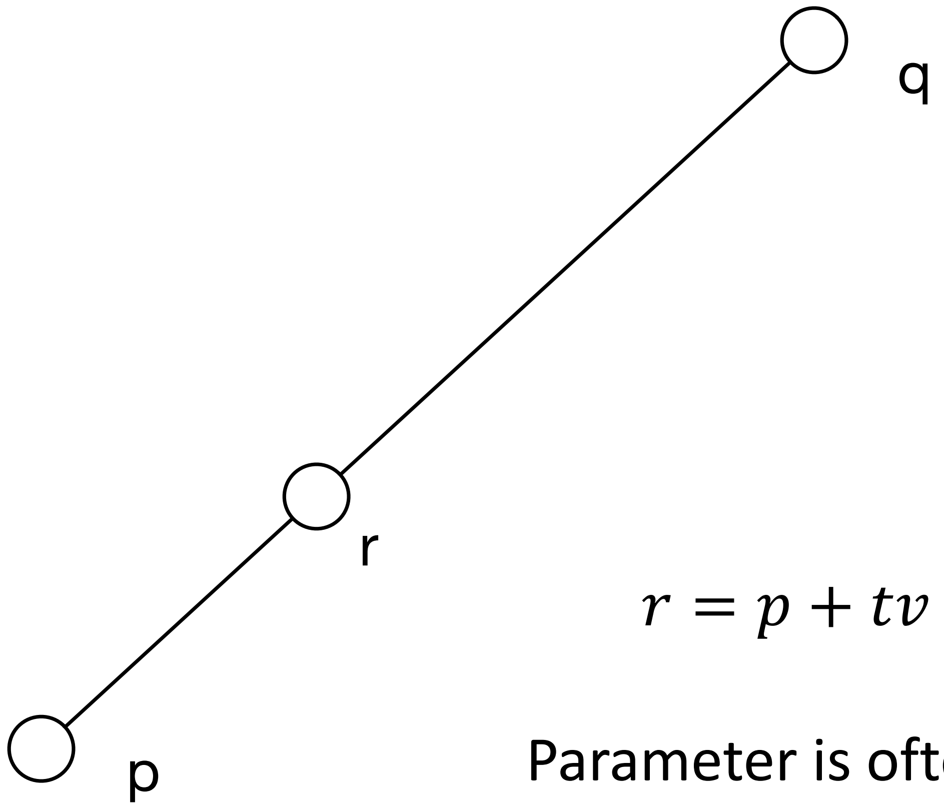
What is …

Midpoint?

Line segment?

Line?

Ray?



$$r = p + tv$$

Parameter is often t for time

Problem 7: Using vector line equation

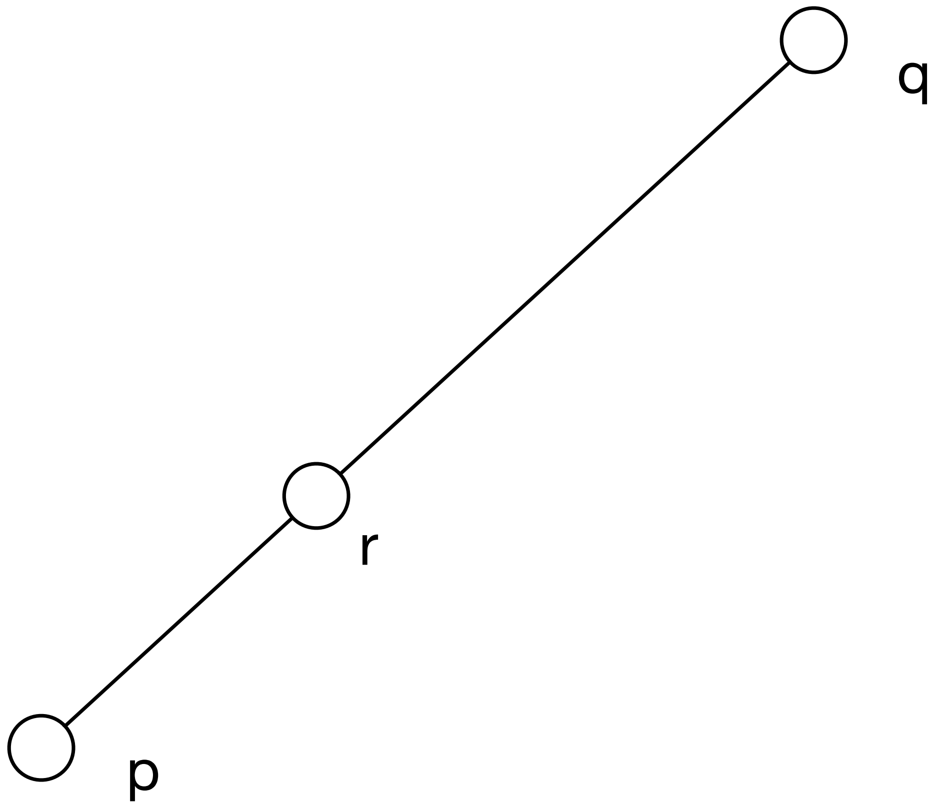
$$r = p + \alpha v$$

Midpoint? $\alpha = 0.5$

Line segment? $\alpha \in [0,1]$

Line? $\alpha \in [-\infty, +\infty]$

Ray? $\alpha \in [0, +\infty]$



Defining line segment by affine combination

$$
\begin{aligned}
r &= p + \alpha v \\
&= p + \alpha(q - p) \\
&= (1 - \alpha)p + \alpha q
\end{aligned}
$$

When $\alpha = 0$?

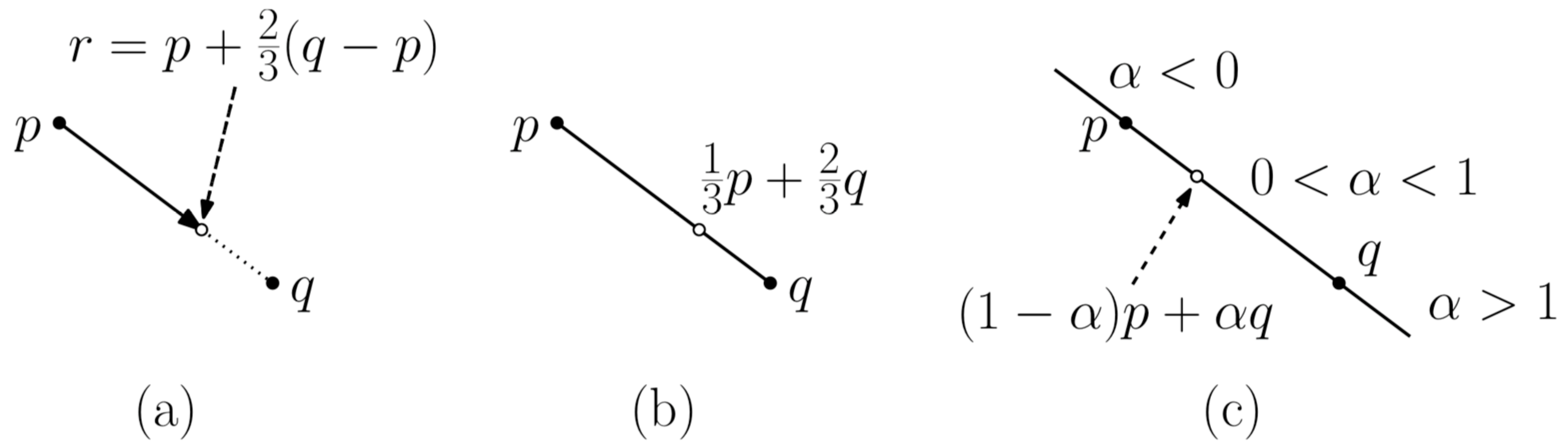
When $\alpha = 1$?

Defining line segment by affine combination

$$\begin{aligned} r &= p + \alpha v \\ &= p + \alpha(q - p) \\ &= (1 - \alpha)p + \alpha q \end{aligned}$$

When $\alpha = 0$? p

When $\alpha = 1$? q



Affine combinations

Given a sequence of points p_1, p_2, ...p_n,
an affine combination is a sum

$$p = \alpha_1 p_1 + \alpha_2 p_2 + \ldots + \alpha_n p_n$$

or

$$p = \sum_{i=1}^{n} \alpha_i p_i$$

With

$$\sum_{i=1}^{n} \alpha_i = 1$$

- This is when you can add points

Affine combinations

Given a sequence of points $p_1, p_2, \ldots, p_n$, an affine combination is a sum

$$p = \alpha_1 p_1 + \alpha_2 p_2 + \ldots + \alpha_n p_n$$

or

$$p = \sum_{i=1}^{n} \alpha_i p_i$$

With

$$\sum_{i=1}^{n} \alpha_i = 1$$

- Is our line equation affine?

$$r = (1 - \alpha)p + \alpha q$$

Convex combinations

Given a sequence of points $p_1, p_2, \ldots, p_n$, a convex combination is a sum

$$p = \alpha_1 p_1 + \alpha_2 p_2 + \ldots + \alpha_n p_n$$

or

$$p = \sum_{i=1}^{n} \alpha_i p_i$$

With

$$\sum_{i=1}^{n} \alpha_i = 1 \textbf{\textit{ and }} \alpha_i \geq 0$$

- What does it mean if our line equation is convex?

$$r = (1 - \alpha)p + \alpha q$$

Problem 8: Centroid of triangle

Given a sequence of points p_1, p_2, …, p_n, an affine combination is a sum

$$p = \alpha_1 p_1 + \alpha_2 p_2 + \ldots + \alpha_n p_n$$

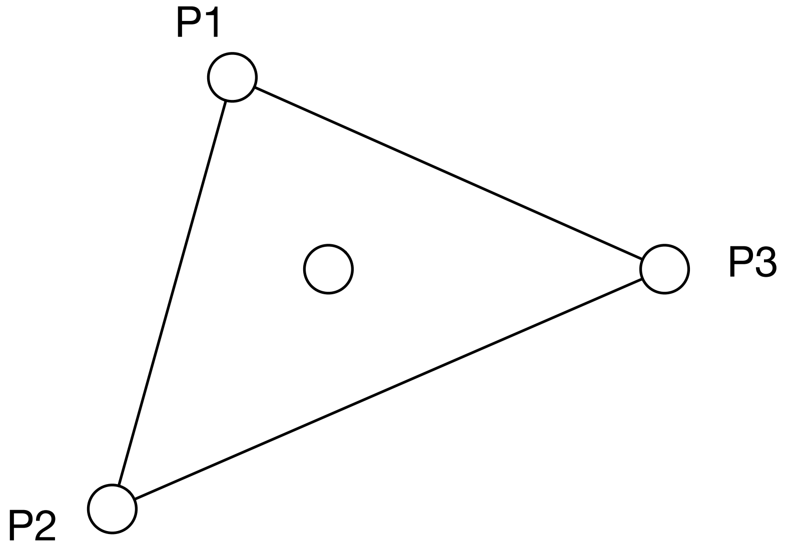
or

$$p = \sum_{i=1}^{n} \alpha_i p_i$$

With

$$\sum_{i=1}^{n} \alpha_i = 1 \; \boldsymbol{and} \; \alpha_i \geq 0$$

- What is the centroid of a triangle?



Problem 8: Centroid of triangle

Given a sequence of points p_1, p_2, …, p_n, a convex combination is a sum

$$p = \alpha_1 p_1 + \alpha_2 p_2 + \ldots + \alpha_n p_n$$

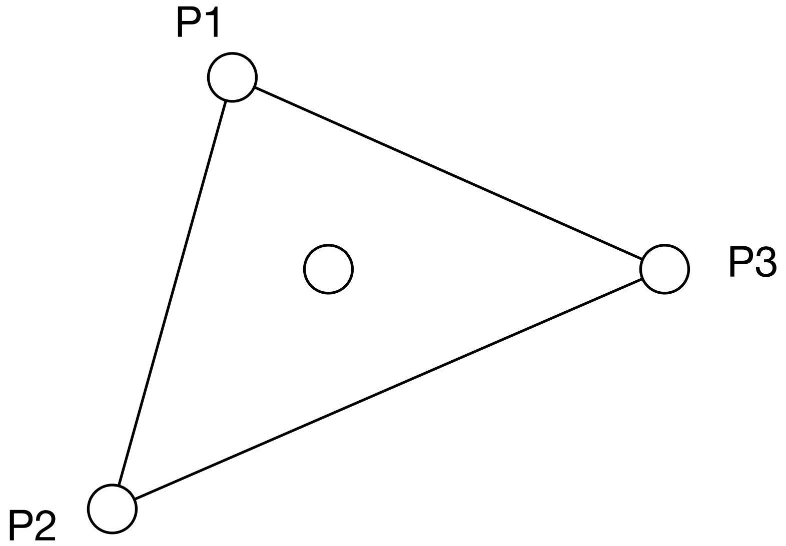
or

$$p = \sum_{i=1}^{n} \alpha_i p_i$$

With

$$\sum_{i=1}^{n} \alpha_i = 1 \textbf{ and } \alpha_i \geq 0$$

- What is the centroid of a triangle?



- 1/3 (p1+p2+p3)

Euclidean geometry: add inner (dot) product

The inner product is an operator that maps two vectors to a scalar. The product of $\vec{u}$ and $\vec{v}$ is denoted commonly denoted $(\vec{u}, \vec{v})$. There are many ways of defining the inner product, but any legal definition should satisfy the following requirements

Positiveness: $(\vec{u}, \vec{u}) \geq 0$ and $(\vec{u}, \vec{u}) = 0$ if and only if $\vec{u} = \vec{0}$.

Symmetry: $(\vec{u}, \vec{v}) = (\vec{v}, \vec{u})$.

Bilinearity: $(\vec{u}, \vec{v} + \vec{w}) = (\vec{u}, \vec{v}) + (\vec{u}, \vec{w})$, and $(\vec{u}, \alpha\vec{v}) = \alpha(\vec{u}, \vec{v})$. (Notice that the symmetric forms follow by symmetry.)

Vector length and normalization

Length: of a vector $\vec{v}$ is defined to be $\sqrt{\vec{v} \cdot \vec{v}}$, and is denoted by $\|\vec{v}\|$ (or as $|\vec{v}|$).

Normalization: Given any nonzero vector $\vec{v}$, define the *normalization* to be a vector of unit length that points in the same direction as $\vec{v}$, that is, $\vec{v}/\|\vec{v}\|$. We will denote this by $\widehat{v}$.

Distance between points: $\mathrm{dist}(p, q) = \|p - q\|$.

Vector length and normalization

Length: of a vector $\vec{v}$ is defined to be $\sqrt{\vec{v} \cdot \vec{v}}$, and is denoted by $\|\vec{v}\|$ (or as $|\vec{v}|$).

Normalization: Given any nonzero vector $\vec{v}$, define the *normalization* to be a vector of unit length that points in the same direction as $\vec{v}$, that is, $\vec{v}/\|\vec{v}\|$. We will denote this by $\widehat{v}$.

Distance between points: $\text{dist}(p, q) = \|p - q\|$.

Question: Does the length of v matter in the equation

$$r = p + \alpha v$$

Vector length and normalization

Length: of a vector $\vec{v}$ is defined to be $\sqrt{\vec{v} \cdot \vec{v}}$, and is denoted by $\|\vec{v}\|$ (or as $|\vec{v}|$).

Normalization: Given any nonzero vector $\vec{v}$, define the *normalization* to be a vector of unit length that points in the same direction as $\vec{v}$, that is, $\vec{v}/\|\vec{v}\|$. We will denote this by $\widehat{v}$.

Distance between points: $\text{dist}(p, q) = \|p - q\|$.

Question: Does the length of v matter in the equation

$$r = p + \alpha v$$

No, as a line or ray

Yes, as a convex combination

Angle between two vectors: cosine law

Angle: between two nonzero vectors $\vec{u}$ and $\vec{v}$ (ranging from 0 to π) is

$$\text{ang}(\vec{u}, \vec{v}) \;=\; \cos^{-1}\left(\frac{\vec{u} \cdot \vec{v}}{\|\vec{u}\|\|\vec{v}\|}\right) \;=\; \cos^{-1}(\widehat{u} \cdot \widehat{v}).$$

This is easy to derive from the law of cosines. Note that this does not provide us with a signed angle. We cannot tell whether $\vec{u}$ is clockwise our counterclockwise relative to $\vec{v}$. We will discuss signed angles when we consider the cross-product.

Orthogonality: $\vec{u}$ and $\vec{v}$ are *orthogonal* (or perpendicular) if $\vec{u} \cdot \vec{v} = 0$.

Problem 9: Angle between p and q?

- p = <0,1>   q= <1,0>

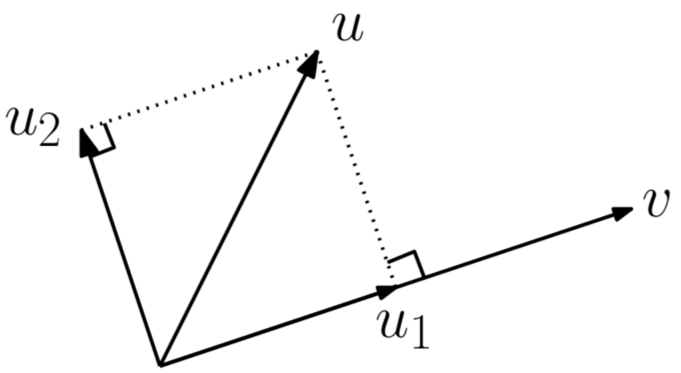
- p = <1,0>   q= <sqrt(2)/2, sqrt(2)/2>

- P = <1,0,1> q= <1,1,0>

Orthogonal projection

Orthogonal projection: Given a vector $\vec{u}$ and a nonzero vector $\vec{v}$, it is often convenient to decompose $\vec{u}$ into the sum of two vectors $\vec{u} = \vec{u}_1 + \vec{u}_2$, such that $\vec{u}_1$ is parallel to $\vec{v}$ and $\vec{u}_2$ is orthogonal to $\vec{v}$.

$$\vec{u}_1 \leftarrow \frac{(\vec{u} \cdot \vec{v})}{(\vec{v} \cdot \vec{v})}\vec{v}, \qquad \vec{u}_2 \leftarrow \vec{u} - \vec{u}_1.$$



Problem 10: Find orthogonal projection

- Given p = <1,1> and q=<1,4>

Given vectors u, v, and w, all of type Vector3, the following operators are supported:

```
u = v + w; // vector addition
u = v - w; // vector subtraction
if (u == v || u != w) { ... } // vector comparison
u = v * 2.0f; // scalar multiplication
v = w / 2.0f; // scalar division
```

You can access the components of a Vector3 using as either using axis names, such as, u.x, u.y, and u.z, or through indexing, such as u[0], u[1], and u[2].

The Vector3 class also has the following members and static functions.

```
float x = v.magnitude; // length of v
Vector3 u = v.normalize; // unit vector in v's direction
float a = Vector3.Angle (u, v); // angle (degrees) between u and v
float b = Vector3.Dot (u, v); // dot product between u and v
Vector3 u1 = Vector3.Project (u, v); // orthog proj of u onto v
Vector3 u2 = Vector3.ProjectOnPlane (u, v); // orthogonal complement
```

Some of the Vector3 functions apply when the objects are interpreted as points. Let p and q be points declared to be of type Vector3. The function Vector3.Lerp is short for *linear interpolation*. It is essentially a two-point special case of a convex combination. (The combination parameter is assumed to lie between 0 and 1.)

```
float b = Vector3.Distance (p, q); // distance between p and q
Vector3 midpoint = Vector3.Lerp(p, q, 0.5f); // convex combination
```

Summary

- After today you should be able to use:

1) Affine data types and operations

   Vector addition, point subtraction, point-vector additions, etc.

2) Affine/convex combinations
3) Euclidean
   1) Dot/inner product
   2) Length, normalization, distance, angle, orthogonality
4) Orthogonal projection
5) Doing it in Unity

Readings

- David Mount's lecture on Geometry and Geometric Programming