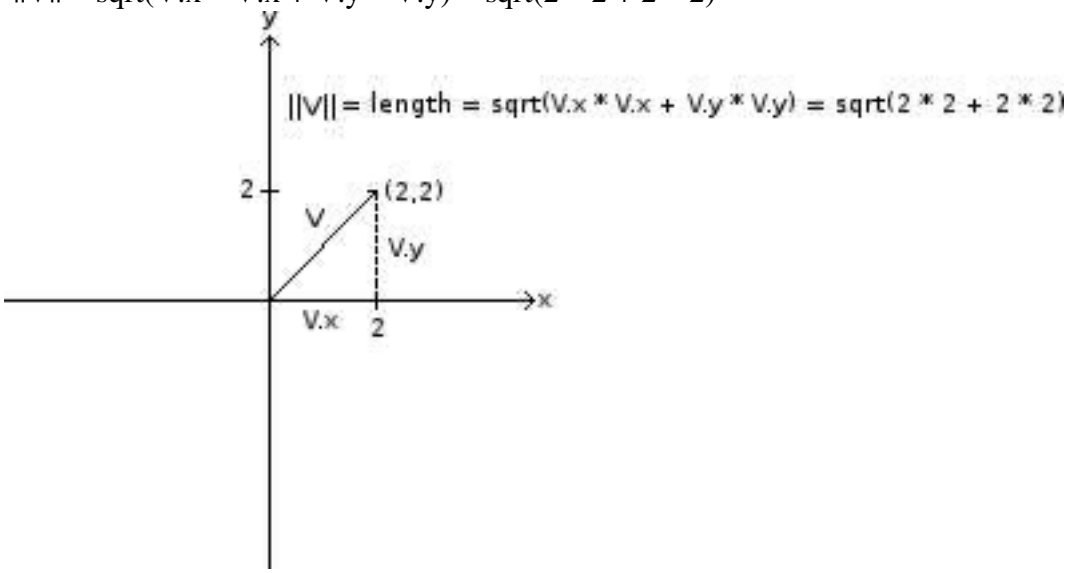
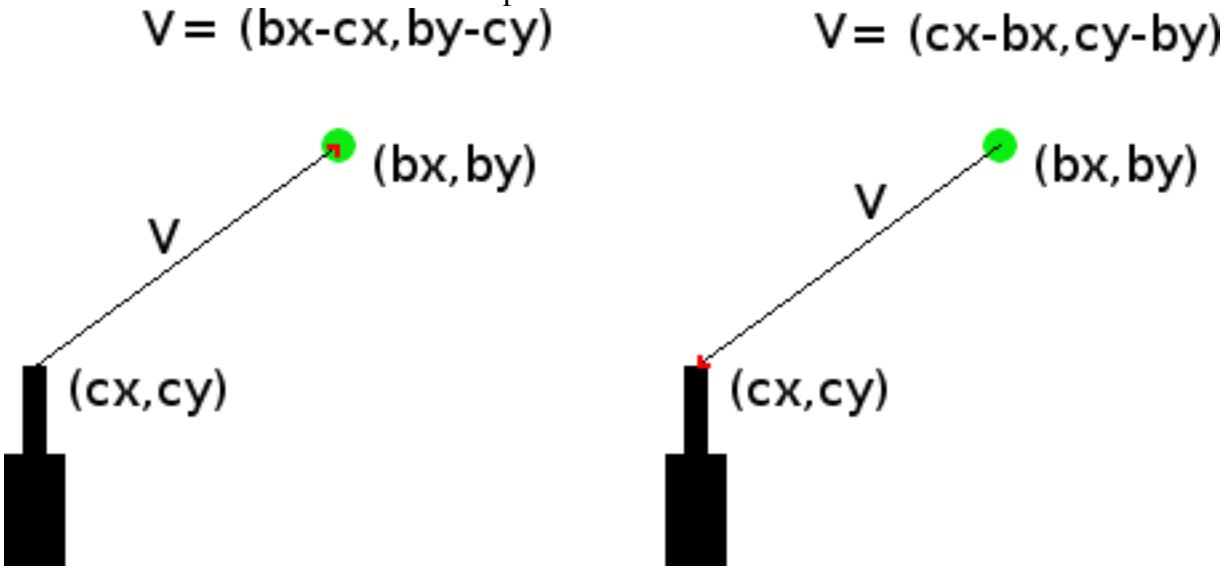
ETGG1801
Game Programming Foundations I
Fall 2011

Vector

- Like a point.
- Represents magnitude (length) and direction instead of position.
  - Forces: gravity, wind, etc.
- Length is found using the Pythagorean Theorem.
  - V = (2, 2)
  - ||V|| = sqrt(V.x * V.x + V.y * V.y) = sqrt(2 * 2 + 2 * 2)



- Find the direction in which to fire a cannon ball.
  - Use the difference between two points to create a vector.

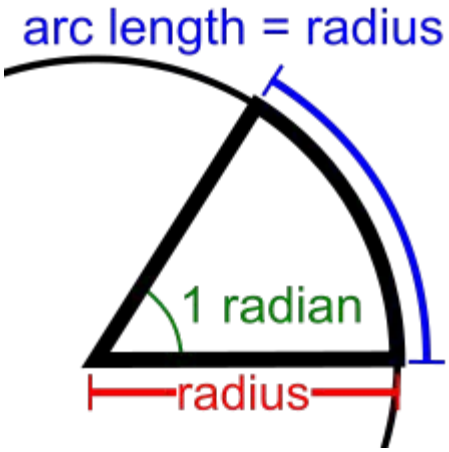


  - The vector's length will be the distance between the two points. Normalize to make its length equal 1.

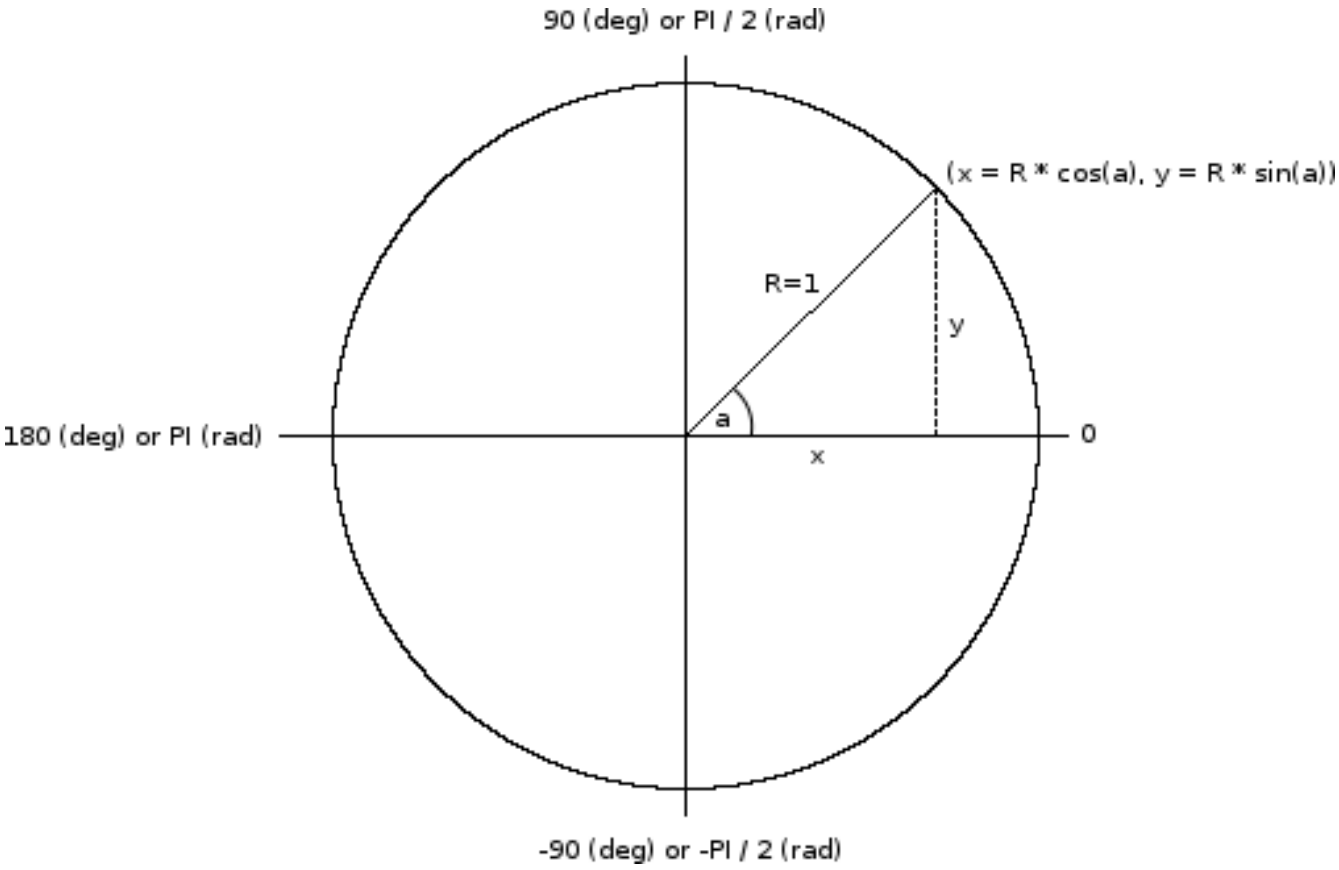
- Normalizing a Vector
  - Makes vector have a length of 1 without changing its direction.
  - Allows us to easily customize the vector's length.
    - Ex: To represent the speed of a bullet.
  - Process:
    - Find vector's length.
    - Divide vector's components (x, y, z, ...) by its length.
      - V = (2, 2)
      - ||V|| = sqrt(2 * 2 + 2 * 2) = sqrt(8)
      - V (normalized) = (V.x / ||V||, V.y / ||V||) = (2 / sqrt(8), 2 / sqrt(8)) = (0.7071, 0.7071)

Rotation

- Trigonometric functions
  - sin = opposite / hypotenuse
  - cos = adjacent / hypotenuse
  - tan = opposite / adjacent
  - Function of an angle
  - SOH-CAH-TOA
- Radian
  - Equal to the angle when arc length equals radius



  - Python's trig functions use radians
    - math.sin
    - math.cos
    - math.tan
  - 1 radian = 180 / PI (degrees)
  - 90 (deg) = 90 * PI / 180 (rad)
  - x = length * cos(angle)
  - y = length * sin(angle)



- Inverse trigonometric functions
  - Cannon example: using direction vector, find angle to rotate the sprite by.
  - arcsin: y = sin(x), arcsin(y) = x
  - arccos
  - arctan
  - Used to find angle
    - V = (2, 2)
    - angle = unknown
    - sin(angle) = opposite / hypotenuse = V.y / ||V|| = 2 / sqrt(8)
    - arcsin(sin(angle)) = arcsin(2 / sqrt(8))
    - angle = arcsin(2 / sqrt(8)) = 45 (deg) or PI / 4 (rad)
  - (2, 2) works, (-2, 2) does not.
    - V = (-2, 2)
    - angle = unknown
    - sin(angle) = opposite / hypotenuse = V.y / ||V|| = 2 / sqrt(8)
    - arcsin(sin(angle)) = arcsin(2 / sqrt(8))
    - angle = arcsin(2 / sqrt(8)) = 45 (deg) or PI / 4 (rad)
      - This is the same answer! It should be 135 (deg).
  - sin(45) has the same value as sin(135).

- Quadrants 1 and 4 (when V.x is positive) work; 2 and 3 (when V.x is negative) do not.
- Apply some logic to fix this:
  - if x < 0:
    - angle = 180 – arcsin(y / length)
  - else:
    - angle = arcsin(y / length)
- Python's math.atan2 function does this for us!