Two-Dimensional Geometry

Vectors

Basic Definition and Explanation

A vector is recording a relative change in position, but isn't fixed to a particular point in space. In two dimensions, a vector simply records a change in x and a change in y. For example, the vector (2, 5) represents moving in a straight line from your current position (x, y) to the position (x+2, y+5). Note that in many times, the difference between a fixed point in space and a vector isn't readily obvious and context has to be used to determine which is which. In the previous explanation, (x, y) is a fixed point and (2, 5) is a vector. Often times, we'll write the vector (2, 5) as 2i + 5j, as i is often referred to as the unit vector in the direction of the positive x axis and j is referred to as the unit vector in the positive y axis. A unit vector is simply a vector with magnitude 1. The magnitude of a vector is simply its length. Thus, the magnitude of the vector ai + bj is $\sqrt{a^2 + b^2}$.

Given two points $A(x_1, y_1)$ and $B(x_2, y_2)$, we can compute the vector AB as $(x_2 - x_1, y_2 - y_1)$. This is the change from pt A to pt B. In essence this vector describes the position of B, relative to A.

Dot Product

The dot product between two vectors returns a scalar (a number). By definition, the dot product between vectors $v_1 = x_1i + y_1j$ and $v_2 = x_2i + y_2j$ is $x_1x_2 + y_1y_2$. It turns out that we can prove that this dot product is ALSO equal to $|v_1||v_2|\cos\theta$, where θ is the angle between the two vectors. In short, if we want to know the angle between two known vectors, we can easily find it via the dot product.

Example: What is the angle between u = 3i + 4j and v = 12i – 5j?

$$u \cdot v = 3(12) + 4(-5) = 16$$
$$u \cdot v = |u||v|cos\theta = \sqrt{3^2 + 4^2}\sqrt{12^2 + 5^2}cos\theta = 65cos\theta$$
$$16 = 65cos\theta, \text{ so } cos\theta = \frac{16}{65}, \text{ and } \theta = cos^{-1}\left(\frac{16}{65}\right) \sim 1.32\ radians$$

Cross Product

The cross product between two vectors returns another vector. By definition, it returns a vector perpendicular to both input vectors with a magnitude equal to the area of the parallelogram defined by both vectors. When we are dealing with 2D geometry, the direction of the cross product is always in the positive or negative z-axis. Thus, this cross product is always (0, 0, z). If $z > 0$, the vector points in the positive z-axis. If $z < 0$, then it points in the negative z-axis. For two dimensions, with vectors $v_1 = x_1i + y_1j$ and $v_2 = x_2i + y_2j$, the cross product is $(0, 0, x_1y_2 - x_2y_1)$. Note that this is also written as $0i + 0j + (x_1y_2 - x_2y_1)k$, since k is the name given to the unit vector in the direction of the positive z-axis. Geometrically, a positive value for the k coefficient of the dot product means that the second vector is counter-clockwise from the first vector by an angle less than π radians. Often times in physics, this is referred to as the "right-hand rule". Basically, in determining the direction of the cross product, you use your right hand, sweeping across from the first vector towards the second and the direction of your thumb is the direction of the resultant vector. We can use the cross product to determine the signed area of a triangle (the sign just tells you which direction you're going in, from side to side), as well as testing if three points are collinear.

Example 1: What is the area of the triangle formed by the following three points: (3, 7), (12, -8), and (9, 16)?

> Create two vectors from (3, 7) to the other two points. These are 9i – 15j and 6i + 9j. The cross product of these is (9(9) – (-15)(6))k = 171k. The absolute value of the k coefficient is the area of the defined parallelogram. Thus, the area of the defined triangle is just half of this, or 85.5.

Example 2: Are the following three points collinear: (3, 8), (19, 40), and (-2, -2)?

> Create two vectors from (3, 8) to the other two points. These are 16i + 32j and -5i – 10j. Their cross product is 16(-10) – (32)(-5) = 0. The points are collinear because sin of the angle between the vectors is 0, consequently that angle is either 0 or π radians!

Line Equation in Two Dimensions

Now that we have a basis in vectors (sorry, bad joke…), we can define the vector equation of a line, which tends to be better to use for programming contest problems than Cartesian equations for lines. A line is defined by a point and a direction. Thus, the vector equation of a line looks like this:

$$r = p_0 + \lambda v$$

where p_0 is a point on the line and v is the directional vector in the direction of the line. Lambda is a parameter such that for any point on the line, there exists a value of lambda that, when plugged in, sets r equal to that point. Consider the following equation of a line:

$$r = (3, -5) + \lambda(2, 1)$$

This essentially means that (3, -5) is on the line and the direction of movement on the line is (2, 1). So, to reach an arbitrary point on the line, start at (3, -5) and move in the direction (2, 1) as far as you want. Logically, this means we can set of a pair of equations solving for the x and y coordinates of any point on the line in terms of lambda:

$$x = 3 + 2\lambda$$
$$y = -5 + 1\lambda$$

These equations are known as parametric equations for the line. For example, if we plug in $\lambda = 3$, we obtain x = 9 and y = -2. This means that (9, -2) is on the line as well. Each unique value we plug in for lambda will produce a different point on the line. For each point on the line, there's a unique value of lambda that creates it.

Line Intersection in Two Dimensions

Now that we know how to express a line in both vector and parametric equations, we can learn how to find the intersection of two lines as follows:

Write out each vector equation. Below r_1 is a line containing (x_1, y_1) in the direction (u_x, u_y) and r_2 is a line containing (x_2, y_2) in the direction (v_x, v_y). Note that we use different parameters because simultaneously, we may care about points on r_1 and r_2 that are created with different parameter values. The only time we would use the same parameters if the parameter represented a time and both equations described the movement of objects in time and the value of the parameter produced the location of each particle at that point in time.

$$r_1 = (x_1, y_1) + \lambda(u_x, u_y)$$
$$r_2 = (x_2, y_2) + \mu(v_x, v_y)$$

Parametrically, for both equations we get:

$$x = x_1 + u_x\lambda \qquad\qquad x = x_2 + v_x\mu$$
$$y = y_1 + u_y\lambda \qquad\qquad y = y_2 + v_y\mu$$

In order for these lines to intersect, there must exist a value for λ and a value for μ, that when plugged into both equations, produces the same point. Thus, for any intersection point, we must have:

$$x_1 + u_x\lambda = x_2 + v_x\mu$$
$$y_1 + u_y\lambda = y_2 + v_y\mu$$

The only unknown quantities are λ and μ. We can rewrite this as a typical pair of linear equations in two variables as follows:

$$u_x\lambda - v_x\mu = x_2 - x_1$$
$$u_y\lambda - v_y\mu = y_2 - y_1$$

By hand, we use many different techniques to solve a system like this and choose the one that minimizes arithmetic. In code, it's probably best to use Kramer's Rule. If our system has a unique solution and is

$$ax + by = c$$
$$dx + ey = f$$

then are solutions are: $x = \frac{\begin{vmatrix} c & b \\ f & e \end{vmatrix}}{\begin{vmatrix} a & b \\ d & e \end{vmatrix}}$ and $y = \frac{\begin{vmatrix} a & c \\ d & f \end{vmatrix}}{\begin{vmatrix} a & b \\ d & e \end{vmatrix}}$. Recall that absolute value bars on the matrices indicates a determinant and for the 2 x 2 case, we have $\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$.

If the denominator of these expressions is zero, this means the system does NOT have unique solutions. Specifically, either the lines are coincidental (describing the same line), or they are parallel and non-intersecting. To determine which is the case, pick two points on one line and one line on the other, and test to see if they are collinear or not.

<u>Line Segment Intersection in Two Dimensions</u>

Do all the work described for regular line intersection and make sure that the vector of movement is the vector of movement from the start point to the end point of the line segment. (You can arbitrarily choose the start and end point unless the problem given specifies a direction of motion where the time variable is meaningful.)

We can quickly see that the intersection point is on one segment if the corresponding parameter of the solution is in between 0 and 1, inclusive. Thus, for the segments to actually intersect, BOTH parameters must be in between 0 and 1 inclusive.

Finally, we have the tricky case of line segments both on coincidental lines. We must take one line and test both endpoints of the other line against the first line. If both endpoints correspond to a parameter greater than 1 or less than 0 on the first line, then there's no intersection. Otherwise, there is.

Example: Do the line segments (3, 9) to (18, -6) and (-4, 16) to (2, 10) intersect?

In solving the appropriate set of equations, we find that the lines are coincidental. Now, our first line, written parametrically is:

$$x = 3 + 15\lambda$$
$$y = 9 – 15\lambda$$

Find the value of λ that corresponds to the point (-4, 16): Just set 3 + 15λ = -4, yielding λ = -7/15.
Find the value of λ that corresponds to the point (2, 10): Just set 3 + 15λ = 2m yielding λ = -1/15.

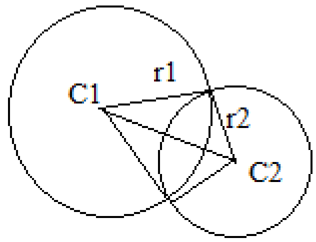
Since both of these are less than 0, there is no intersection.

If we move the second point of the second segment to be (4, 8), then its λ = 1/15 and there would be an intersection.

Alternatively, if we move the second point of the second segment to be (20, -8), then its λ = 17/15 would still indicate an intersection because both values aren't either less than 0 or greater than 1.

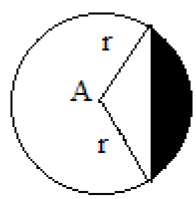
Circle-Circle Intersection

Let circles C_1 and C_2 have radii r_1 and r_2, respectively. If the distance between C_1 and C_2 exceeds $r_1 + r_2$, there is no intersection. If this distance is equal, then there's one intersection, which is on both circles and the line segment between both centers. If this distance is less then we have two intersections. Here is a picture:



Since we know r_1 and r_2 and the distance between the two centers, we know all three of sides of the two congruent triangles in the picture. This allows for us to solve for all three angles in these triangles. Using the atan2 function, we can get the directional angle from C_1 to C_2, add/subtract to it the appropriate angle in the triangle to get the new directional angle from C_1 to each intersection point. We can move along each of these vectors a distance of r_1 to get the solution points.

If we want to fine the area of that "eye-shaped" region, we can calculate the sum of two sector minus triangle calculations. Here is an illustration of one such calculation:



The area of the sector is $\frac{A}{2}r^2$ while the area of the triangle is $\frac{1}{2}r^2 sin\ A$. Thus, the shaded area is the difference of these two: $\frac{r^2}{2}(A - sinA)$. This difference of area also cleverly proves that A > sin A for 0 < A < π. Note that once we know angle A, we can compute the length of the chord on the circle using the law of cosines. In particular, this length is equal to $r\sqrt{2(1 - cosA)}$.

Circle-Line Intersection

Given an equation of a circle $(x - c_x)^2 + (y - c_y)^2 = r^2$ and a line in parametric form with the equation

$$x = x_1 + \lambda d_x$$
$$y = y_1 + \lambda d_y$$

we can simply plug in our arbitrary expression for x and y for any point on the line into the circle equation:

$$(x_1 + \lambda d_x - c_x)^2 + (y_1 + \lambda d_y - c_y)^2 = r^2$$

Noting that the only thing unknown in this equation is λ, we see that once we simplify, we can rewrite this equation as a quadratic in λ:

$$(d_x^2 + d_y^2)\lambda^2 - 2(d_x(x_1 - c_x) + d_y(y_1 - c_y))\lambda + (x_1 - c_x)^2 + (y_1 - c_y)^2 - r^2 = 0$$

If this equation has no solutions, there is no intersection. If it has one solution there is one solution, which is a point of tangency, otherwise, there are two solutions. Once we get the appropriate values of λ, we can plug these back into the line equation to get the corresponding points of intersection on the line with the circle. If our input line was a segment, we would have check to see if either parameter for the intersection was in between 0 and 1.