Hình học tính toán phần 2: Sự giao nhau của đường thẳng và các ứng dụng

Hình học tính toán phần 2: Sự giao nhau của đường thẳng và các ứng dụng

Tác giả:

- Lê Minh Hoàng - Phổ thông Năng khiếu, ĐHQG-HCM

Reviewer:

- Trần Quang Lộc - ITMO University
- Hoàng Xuân Nhật - Đại học Khoa học Tự nhiên, ĐHQG-HCM
- Hồ Ngọc Vĩnh Phát - Đại học Khoa học Tự nhiên, ĐHQG-HCM

Trong phần 1, chúng ta đã tìm hiểu cách sử dụng vector để giải các bài toán hình học. Bây giờ chúng ta sẽ học cách sử dụng một vài kiến thức đại số tuyến tính cơ bản để tìm giao điểm của đường thẳng, sau đó áp dụng để giải quyết một số bài toán khác.

Lưu ý: một số hình ảnh được chụp từ Desmos, và đều có link Desmos tương ứng ở dưới mỗi hình, các bạn có thể nhấn vào link để tương tác với hình và các tham số.

Giao điểm của hai đường thẳng

Một trong những bài toán con phổ biến nhất trong các bài toán hình học là giao điểm đường thẳng. Mặc dù phổ biến nhưng nhiều người vẫn gặp rắc rối với nó.

Đầu tiên, ta có câu hỏi nhỏ là: đường thẳng được cho dưới dạng nào? và chúng ta muốn sử dụng ở dạng nào? Ở trường hợp lý tưởng thì đường thẳng sẽ ở dạng $Ax + By = C$, với A, B, C là các hệ số xác định đường thẳng. Tuy nhiên, ta hiếm khi được cho đường thẳng ở dạng này, nhưng ta có thể dễ dàng có được từ hai điểm cho trước. Ví dụ có hai điểm **phân biệt** (x_1, y_1) và (x_2, y_2), và để tìm A, B, C cho phương trình trên, ta đặt:

$$\begin{cases} A = y_2 - y_1 \\ B = x_1 - x_2 \\ C = Ax_1 + By_1 & = (y_2 - y_1)x_1 + (x_1 - x_2)y_1 \\ & = x_1y_2 - x_1y_1 + x_1y_1 - x_2y_1 \\ & = x_1y_2 - x_2y_1 \end{cases}$$

Bất kể đường thẳng được cho dưới dạng nào, ta luôn có thể chọn được hai điểm phân biệt thuộc đường thẳng, sau đó tính A, B, C.

Tiếp theo, giả sử ta có hai đường thẳng, được cho bởi hai phương trình:

$$\begin{cases} A_1x + B_1y = C_1 \\ A_2x + B_2y = C_2 \end{cases}$$

Để tìm giao điểm của hai đường thẳng, ta chỉ cần giải hệ hai phương trình với hai ẩn x, y:

```
double det = A1 * B2 - A2 * B1;
if (det == 0) {
    // Lines are parallel or coincident
    if (A1 * C2 == A2 * C1) {
        // Lines are coincident
    }
    else {
        // Lines are parallel
    }
}
else {
    double x = (B2 * C1 - B1 * C2) / det;
    double y = (A1 * C2 - A2 * C1) / det;
}
```

Để biết được công thức ở đoạn code trên từ đâu ra, ta nhân phương trình thứ nhất với B_2, và nhân phương trình thứ hai với B_1:

$$\begin{cases} A_1B_2x + B_1B_2y = B_2C_1 \\ A_2B_1x + B_1B_2y = B_1C_2 \end{cases}$$

Kế tiếp, lấy phương trình thứ nhất trừ phương trình thứ hai:

$$A_1B_2x - A_2B_1x = B_2C_1 - B_1C_2$$

Cuối cùng, chia cả hai vế cho $A_1B_2 - A_2B_1$, ta sẽ có phương trình giải x:

$$x = \frac{B_2C_1 - B_1C_2}{A_1B_2 - A_2B_1}$$

Phương trình giải y thu được bằng biến đổi tương tự:

$$y = \frac{A_1C_2 - A_2C_1}{A_1B_2 - A_2B_1}$$

Như vậy ta sẽ có được toạ độ giao điểm của hai đường thẳng, nhưng sẽ thế nào nếu đây là hai đoạn thẳng? Trong trường hợp này, ta cần kiểm tra xem giao điểm tìm được có nằm trên hai đoạn thẳng hay không.
Nếu đoạn thẳng nối hai điểm (x_1, y_1) và (x_2, y_2), để kiểm tra xem (x, y) có thuộc đoạn thẳng hay không, ta chỉ cần kiểm tra $min(x_1, x_2) \leq x \leq max(x_1, x_2)$ và làm tương tự cho y.

Ta cũng nên cẩn thận với vấn đề về độ chính xác của số thực. Nếu giao điểm nằm ngay trên đầu mút của đoạn thẳng, hoặc nếu đoạn thẳng nằm ngang hoặc thẳng đứng, một phép so sánh tầm thường có thể có vấn đề.

Trong những trường hợp đó, ta có thể thực hiện so sánh với một giá trị sai số nào đó (thường là 10^{-9}) hoặc sử dụng phân số.

Ngoài ra, ta còn có thể sử dụng tích có hướng để kiểm tra hai đoạn thẳng giao nhau với ưu điểm là không phụ thuộc vào sai số khi tọa độ các đỉnh đều là số nguyên.

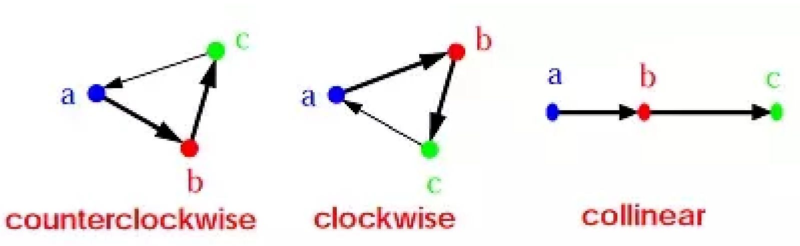
Kiểm tra giao điểm của 2 đoạn thẳng (sử dụng tích có hướng)

CW và CCW

Nhắc lại phần 1: Với góc α thỏa mãn $0^\circ < \alpha < 180^\circ$ thì $\sin(\alpha) > 0$ nên nếu góc ngược chiều kim đồng hồ $\theta < 180^\circ$ thì tích có hướng **dương**, ngược lại tích có hướng **âm**.

Để biết thứ tự của bộ 3 điểm A, B, C, ta tính tích có hướng $\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AC}}$:

- Nếu $\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AC}} > 0$ thì A, B, C ngược chiều kim đồng hồ (CCW).
- Nếu $\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AC}} < 0$ thì A, B, C cùng chiều kim đồng hồ (CW).
- Nếu $\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AC}} = 0$ thì A, B, C thẳng hàng.

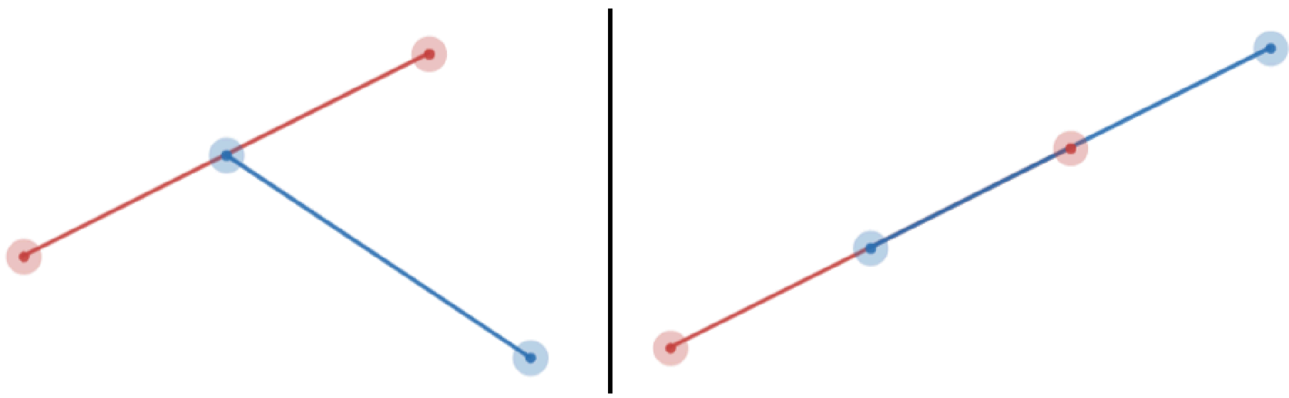


Kiểm tra giao điểm của 2 đoạn thẳng

Tồn tại 3 điểm thẳng hàng

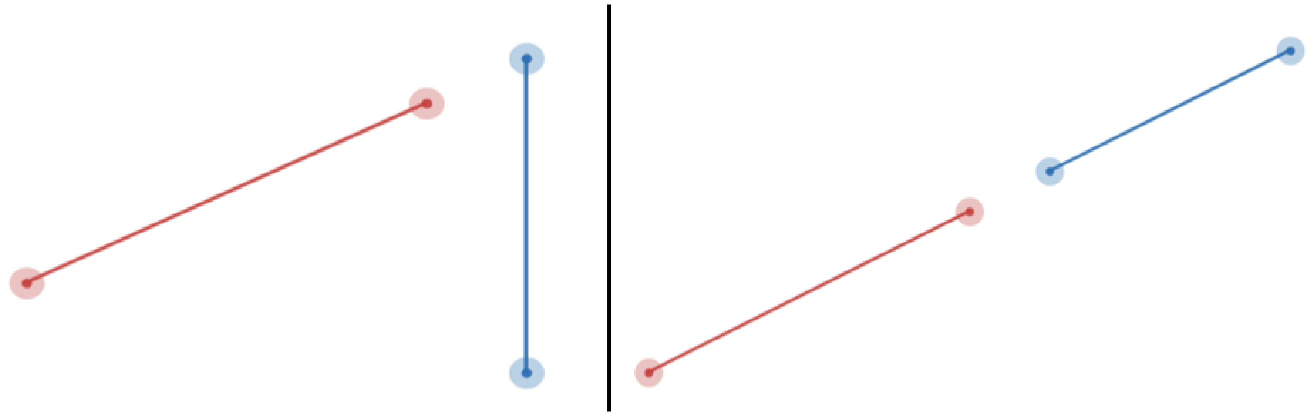
Nếu tồn tại 3 trong 4 điểm đầu mút thẳng hàng, ta kiểm tra xem có tồn tại đầu mút của đoạn thẳng này thuộc đoạn thẳng kia hay không:

- Nếu có thì rõ ràng là 2 đoạn thẳng giao nhau tại ít nhất 1 điểm (tại đầu mút vừa xét).



- Nếu không thì rõ ràng là 2 đoạn thẳng không thể giao nhau.

|



|

Không tồn tại 3 điểm thẳng hàng

Nếu không tồn tại 3 trong 4 điểm đầu mút thẳng hàng thì 2 đoạn thẳng AB và CD giao nhau khi:

- C và D nằm khác phía đối với đường thẳng AB và
- A và B nằm khác phía đối với đường thẳng CD.

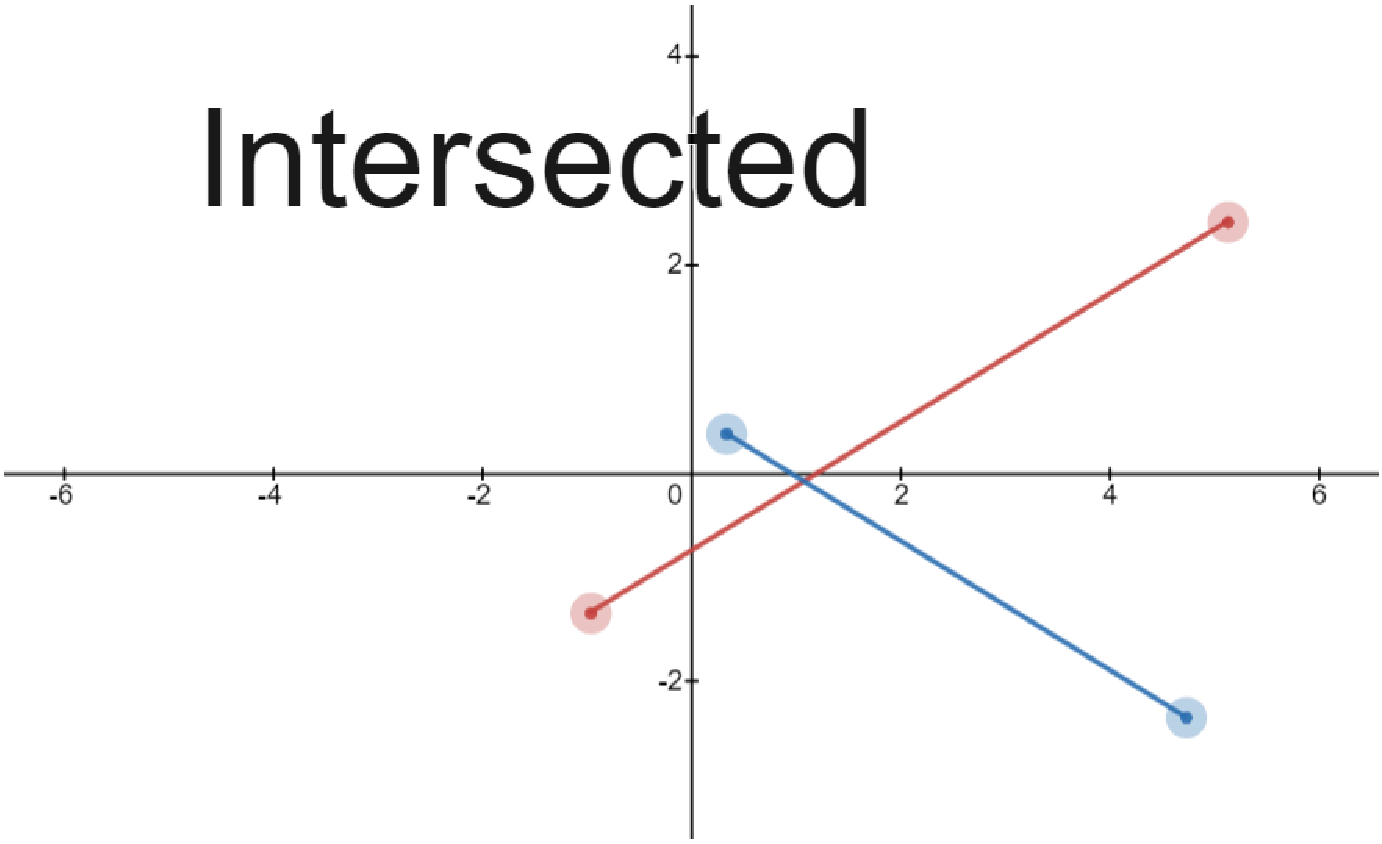
Để C và D nằm khác phía đối với đường thẳng AB thì:

- A, B, C ngược chiều kim đồng hồ và A, B, D cùng chiều kim đồng hồ *hoặc*
- A, B, C cùng chiều kim đồng hồ và A, B, D ngược chiều kim đồng hồ.

$$\implies (\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AC}}) \cdot (\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AD}}) < 0$$

Từ đó, ta có hệ sau:

$$\begin{cases} (\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AC}}) \cdot (\overrightarrow{\mathrm{AB}} \times \overrightarrow{\mathrm{AD}}) < 0 \\ (\overrightarrow{\mathrm{CD}} \times \overrightarrow{\mathrm{CA}}) \cdot (\overrightarrow{\mathrm{CD}} \times \overrightarrow{\mathrm{CB}}) < 0 \end{cases}$$



Nhấn vào **đây** để tương tác với hình trên Desmos.

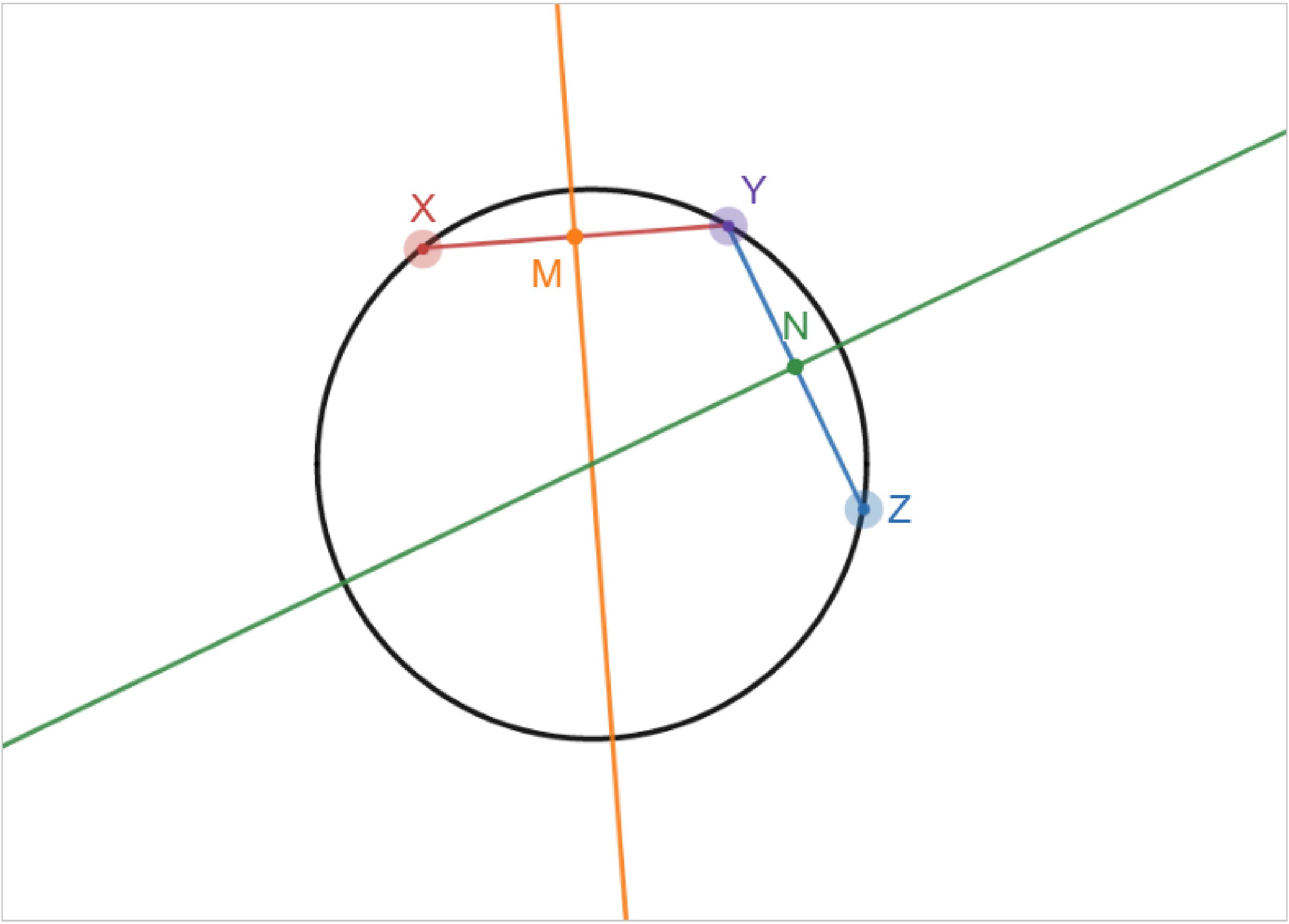
```
const double eps = 1e-9;
int sign(double x) {
    if (x > eps) return 1;
    if (x < -eps) return -1;
    return 0;
}
double cross(Vec AB, Vec AC) {
    return AB.x * AC.y - AC.x * AB.y;
}
double dot(Vec AB, Vec AC) {
    return AB.x * AC.x + AB.y * AC.y;
}
bool intersect(Point A, Point B, Point C, Point D) {
    int ABxAC = sign(cross(B - A, C - A));
    int ABxAD = sign(cross(B - A, D - A));
    int CDxCA = sign(cross(D - C, A - C));
    int CDxCB = sign(cross(D - C, B - C));
    if (ABxAC == 0 || ABxAD == 0 || CDxCA == 0 || CDxCB == 0) {
        // C on segment AB if ABxAC = 0 and CA.CB <= 0
        if (ABxAC == 0 && sign(dot(A - C, B - C)) <= 0) return true;
        if (ABxAD == 0 && sign(dot(A - D, B - D)) <= 0) return true;
        if (CDxCA == 0 && sign(dot(C - A, D - A)) <= 0) return true;
        if (CDxCB == 0 && sign(dot(C - B, D - B)) <= 0) return true;
        return false;
```

```
    }
    return (ABxAC * ABxAD < 0 && CDxCA * CDxCB < 0);
}
```

Tìm đường tròn đi qua 3 điểm

Từ 3 điểm không thẳng hàng, có duy nhất một đường tròn đi qua 3 điểm đó. Nhưng làm thế nào để tìm được tâm của nó? Vấn đề này hoá ra là một ứng dụng đơn giản của bài toán giao điểm đường thẳng.

Chúng ta sẽ tìm đường trung trực của 2 đoạn XY và YZ, sau đó tìm giao điểm của hai đường này, điểm đó chính là tâm của đường tròn.



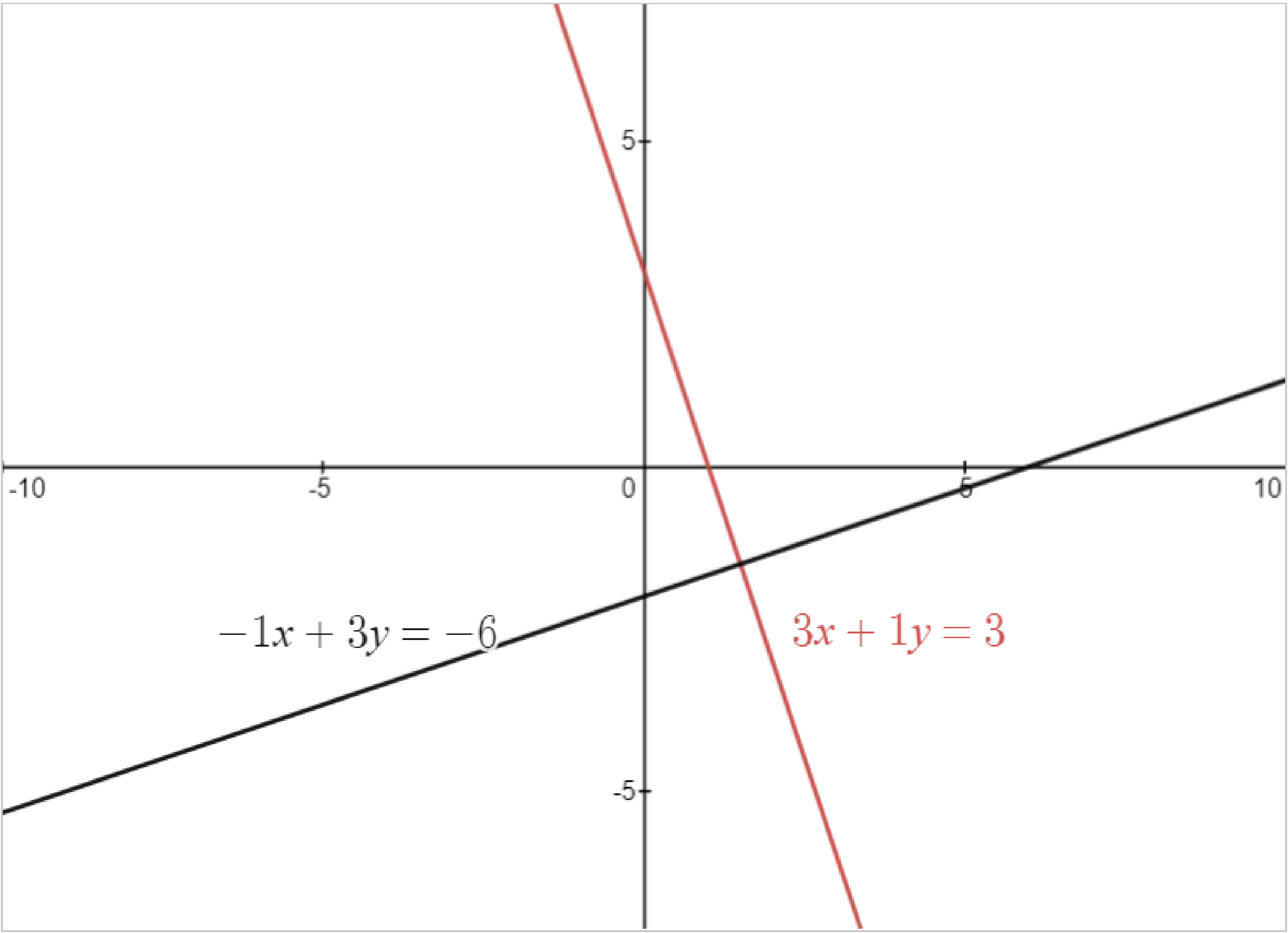
Nhấn vào **đây** để tương tác với hình trên Desmos.

Trong hình học phẳng, đường trung trực của một đoạn thẳng là đường **vuông góc** với đoạn thẳng **tại trung điểm** của đoạn thẳng đó.

Các bước để tìm đường trung trực của đoạn XY:

- **Bước 1:** tìm phương trình đường thẳng XY, giả sử có dạng $Ax + By = C$.
- **Bước 2:** tìm trung điểm M của đoạn XY bằng cách lấy trung bình cộng của 2 hoành độ và trung bình cộng của 2 tung độ.

- **Bước 3:** viết phương trình đường thẳng của đường thẳng vuông góc với đường thẳng XY có dạng là $-Bx + Ay = D$.



Nhấn vào **đây** để tương tác với hình trên Desmos.

- **Bước 4:** ta thay tọa độ của trung điểm M vào phương trình đường thẳng ở bước 3 để tìm D và xác định đường trung trực.

Ví dụ

Cho 2 điểm $X(2, -3)$ và $Y(1, 0)$, để tìm đường trung trực của đoạn XY, ta thực hiện như sau:

- **Bước 1:** Tìm phương trình đường thẳng XY, ta đặt:

$$\begin{cases} A = Y_y - X_y = 0 - (-3) = 3 \\ B = X_x - Y_x = 2 - 1 = 1 \\ C = AX_x + BX_y = 3 \cdot 2 + 1 \cdot (-3) = 3 \end{cases}$$

$\Longrightarrow$ phương trình đường thẳng XY có dạng: $3x + y = 3$

- **Bước 2:** Đặt M là trung điểm của đoạn XY, ta có:

$$\begin{cases} M_x = \dfrac{X_x + Y_x}{2} = \dfrac{2+1}{2} = 1,5 \\ M_y = \dfrac{X_y + Y_y}{2} = \dfrac{-3+0}{2} = -1,5 \end{cases}$$

- **Bước 3:** Phương trình đường thẳng của đường thẳng vuông góc với đường thẳng XY có dạng: $-x + 3y = D$
- **Bước 4:** Thay tọa độ của trung điểm M vào phương trình $-x + 3y = D$:

$$-(1,5) + 3 \cdot (-1,5) = D \Rightarrow D = -6$$

Vậy phương trình đường trung trực của đoạn XY là: $-x + 3y = -6$

Làm tương tự cho đoạn YZ, chúng ta sẽ có hai phương trình của hai đường trung trực, và có thể tìm giao điểm của chúng như đã đề cập ở trên.

```
struct Point {
    double x, y;
    Point() { x = y = 0.0; }
    Point(double x, double y) : x(x), y(y) {}

    Point operator + (const Point &a) const { return Point(x + a.x, y + a.y);
    Point operator - (const Point &a) const { return Point(x - a.x, y - a.y);
    Point operator * (double k) const { return Point(x * k, y * k); }
    Point operator / (double k) const { return Point(x / k, y / k); }
};

struct Line { // Ax + By = C
    double a, b, c;
    Line(double a = 0, double b = 0, double c = 0) : a(a), b(b), c(c) {}
    Line(Point A, Point B) {
        a = B.y - A.y;
        b = A.x - B.x;
        c = a * A.x + b * A.y;
    }
};

Line Perpendicular_Bisector(Point A, Point B) {
    Point M = (A + B) / 2;
    Line d = Line(A, B);
    // the equation of a perpendicular line has the form: -Bx + Ay = D
    double D = -d.b * M.x + d.a * M.y;
    return Line(-d.b, d.a, D);
}
```

Phép đối xứng

Để lấy đối xứng một điểm X qua một đường thẳng (trục đối xứng), ta tìm giao điểm Y của trục đối xứng và đường thẳng vuông góc với trục đối xứng đi qua X, sau đó lấy X' đối xứng với X qua Y.

Ví dụ

Cho điểm $X(1, -3)$ và đường thẳng $(d): 4x - 3y = -5$, để tìm điểm X' đối xứng với X qua (d), ta thực hiện như sau:

- **Bước 1**: Gọi đường thẳng đi qua X và vuông góc với trục đối xứng có dạng: $(d'): 3x + 4y = D$.
- **Bước 2**: Để tìm D, ta chỉ cần thay toạ độ của X vào phương trình:

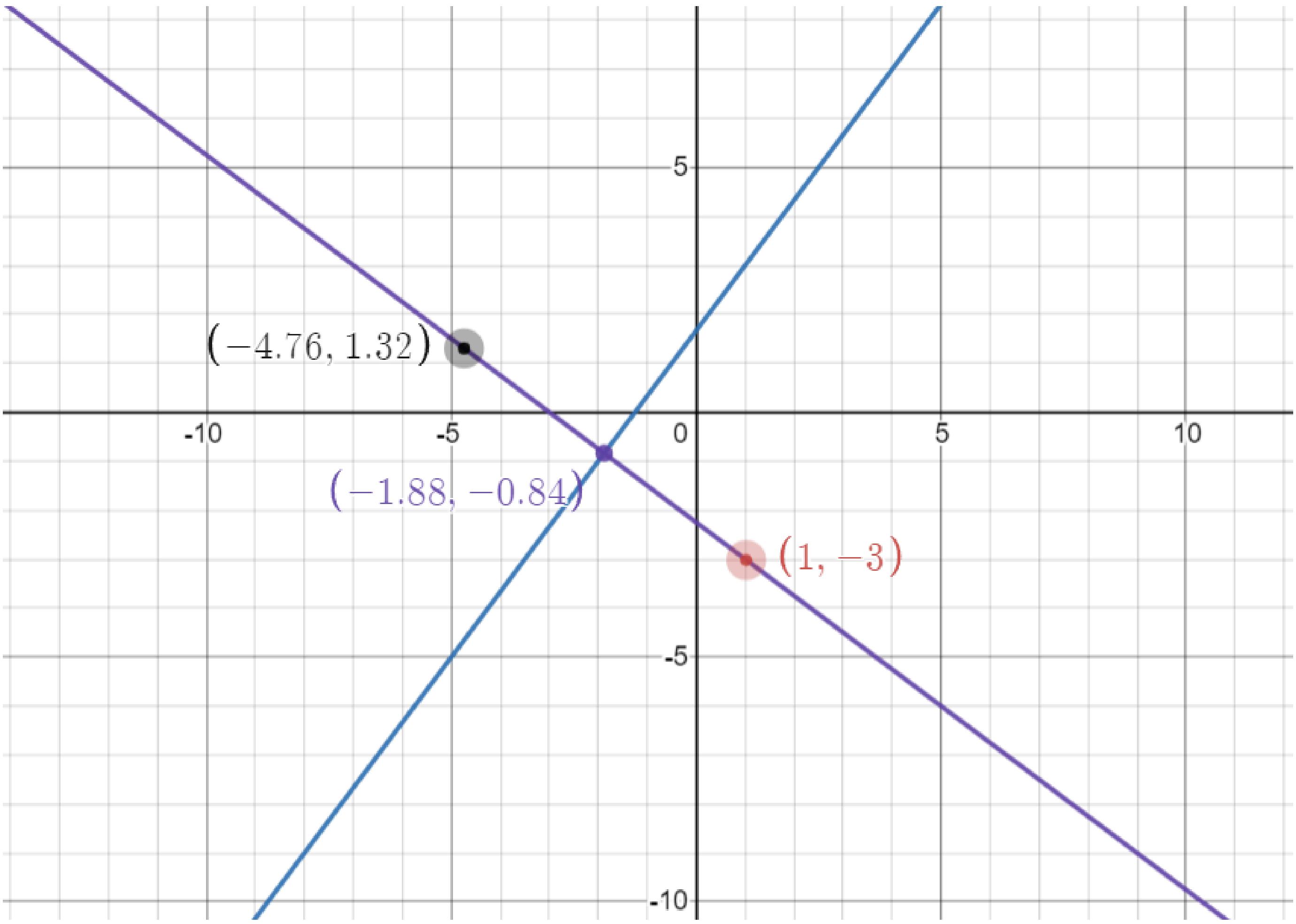
$$3 \cdot 1 + 4 \cdot (-3) = D \iff D = -9 \Rightarrow (d'): 3x + 4y = -9$$

- **Bước 3**: xác định giao điểm Y của hai đường (d) và (d'):

$$\begin{cases} Y_x = \dfrac{B_2C_1 - B_1C_2}{A_1B_2 - A_2B_1} = \dfrac{4 \cdot (-5) - (-3) \cdot (-9)}{4 \cdot 4 - 3 \cdot (-3)} = \dfrac{-47}{25} = -1.88 \\ Y_y = \dfrac{A_1C_2 - A_2C_1}{A_1B_2 - A_2B_1} = \dfrac{4 \cdot (-9) - 3 \cdot (-5)}{4 \cdot 4 - 3 \cdot (-3)} = \dfrac{-21}{25} = -0.84 \end{cases}$$

- **Bước 4**: xác định X' đối xứng với X qua Y bằng công thức: $X' = 2Y - X$ (Y là trung điểm X và X' nên $X + X' = 2Y$).

$$\begin{cases} X'_x = 2Y_x - X_x = 2 \cdot (-1.88) - 1 = -4.76 \\ X'_y = 2Y_y - X_y = 2 \cdot (-0.84) - (-3) = 1.32 \end{cases}$$



Nhấn vào **đây** để tương tác với hình trên Desmos.

```
struct Line { // Ax + By = C
    double a, b, c;
    Line(double a = 0, double b = 0, double c = 0) : a(a), b(b), c(c) {}
};

Point intersect(Line d1, Line d2) {
    double det = d1.a * d2.b - d2.a * d1.b;
    // det != 0 because d1 is perpendicular to d2
    return Point((d2.b * d1.c - d1.b * d2.c) / det, (d1.a * d2.c - d2.a * d1.c
}

Point Symmetry(Point X, Line d) {
    // the equation of a perpendicular line has the form: -Bx + Ay = D
    double D = -d.b * X.x + d.a * X.y;
    Line d2 = Line(-d.b, d.a, D);
    Point Y = intersect(d, d2);
    Point X2 = Point(2 * Y.x - X.x, 2 * Y.y - X.y);
    return X2;
}
```

Phép quay

Cho điểm $A(x, y)$, để quay điểm A **ngược chiều kim đồng hồ** một góc θ quanh **gốc tọa độ**, ta đơn giản sử dụng công thức:

$$\begin{cases} x' = x\cos\theta - y\sin\theta \\ y' = x\sin\theta + y\cos\theta \end{cases}$$

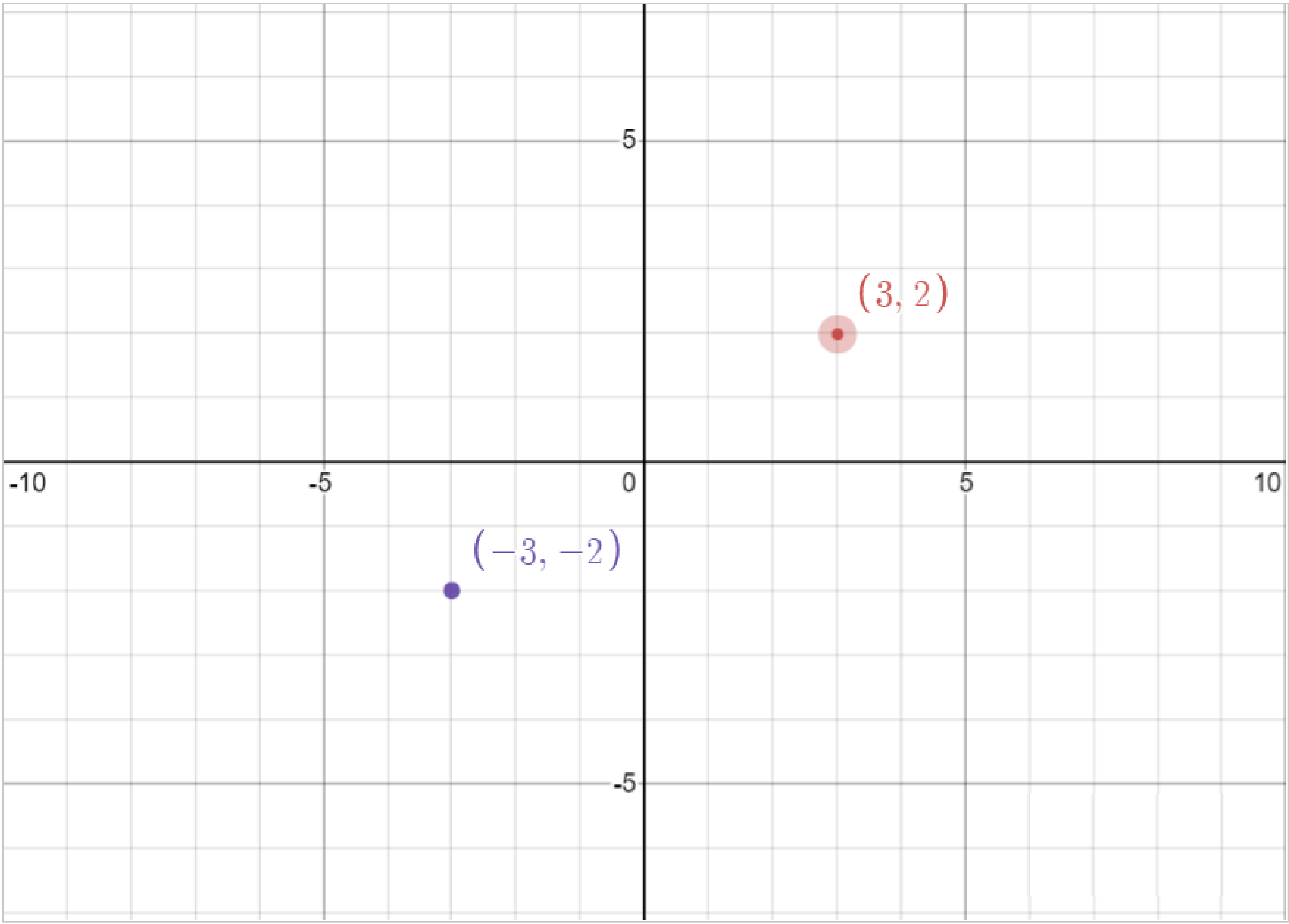
Lưu ý: vì các ngôn ngữ lập trình sử dụng radian(rad) làm đơn vị chuẩn khi làm việc với các hàm số lượng giác nên ở trong desmos, mình sử dụng đơn vị của số đo góc là radian thay vì độ(°).

Công thức chuyển đổi giữa radian và độ:

$$\pi rad = 180^\circ \implies \begin{cases} radian = \dfrac{độ \cdot \pi}{180} \\ độ = \dfrac{radian \cdot 180}{\pi} \end{cases}$$

Bảng chuyển đổi một số giá trị thường dùng:

Độ	0°	15°	30°	45°	60°	75°	90°	180°
Radian	0	$\frac{\pi}{12}$	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{5\pi}{12}$	$\frac{\pi}{2}$	π



Nhấn vào **đây** để tương tác với hình trên Desmos.

Để quay điểm A quanh một điểm C khác không phải gốc tọa độ, ta tịnh tiến hệ tọa độ sao cho C trùng với gốc tọa độ, quay bằng công thức trên rồi tịnh tiến hệ tọa độ về vị trí ban đầu.

Ví dụ

Cho 2 điểm $A(1,4)$ và $C(2,2)$, để quay A ngược chiều kim đồng hồ 1 góc 45° quanh C, ta thực hiện như sau:

- **Bước 1:** tịnh tiến hệ tọa độ sao cho C trùng với gốc tọa độ. Lúc này, điểm A có tọa độ mới là $A' = (1-2, 4-2) = (-1, 2)$.
- **Bước 2:** quay A' ngược chiều kim đồng hồ 1 góc 45° quanh gốc tọa độ được điểm B':

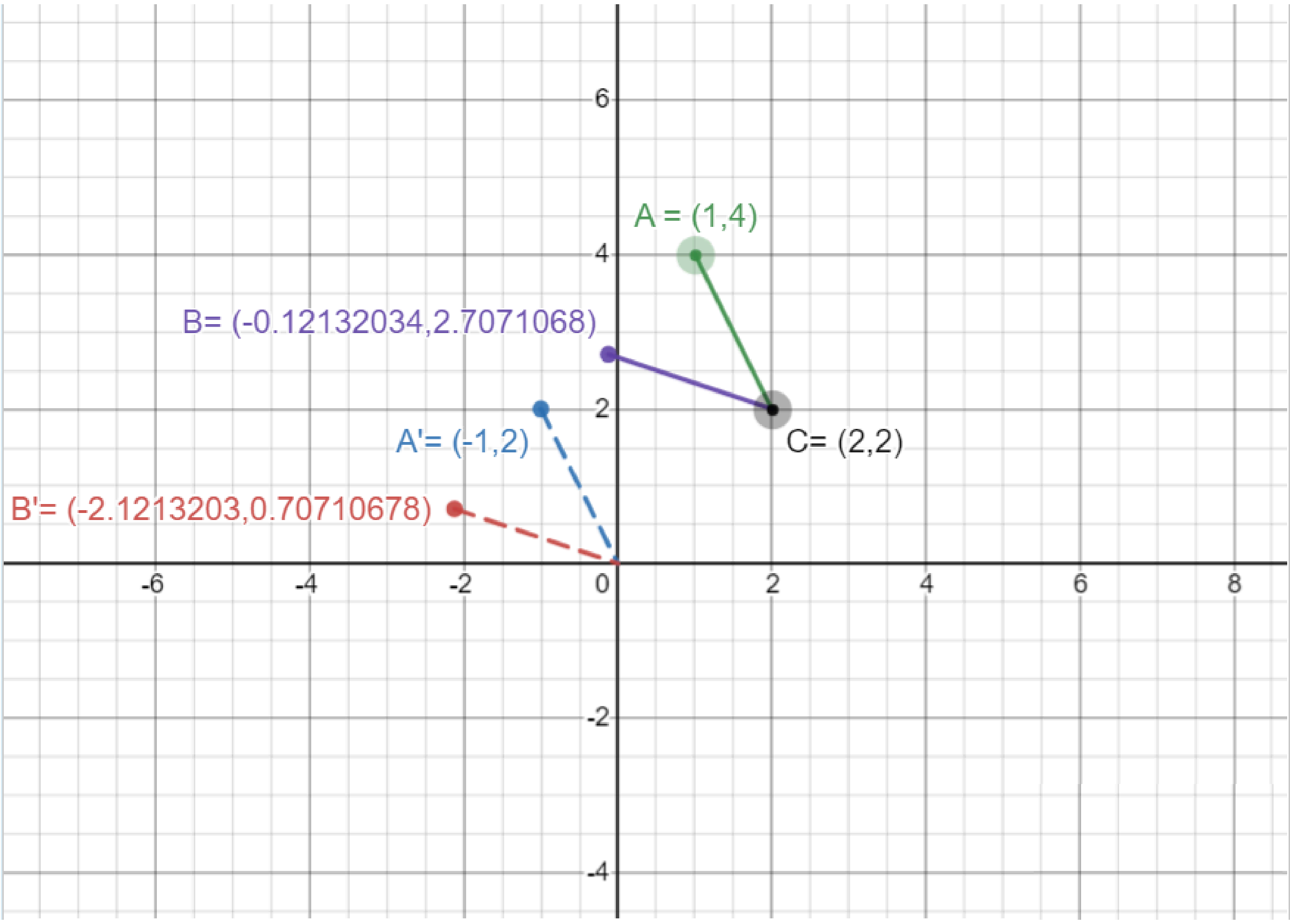
$$\begin{cases} x_{B'} = -1 \cdot \cos 45^\circ - 2 \cdot \sin 45^\circ = -\dfrac{3\sqrt{2}}{2} \\ y_{B'} = -1 \cdot \sin 45^\circ + 2 \cdot \cos 45^\circ = \dfrac{\sqrt{2}}{2} \end{cases}$$

- **Bước 3:** tịnh tiến hệ tọa độ về vị trí ban đầu. Điểm B' có tọa độ mới là:

$$B = \left(-\frac{3\sqrt{2}}{2} + 2, \frac{\sqrt{2}}{2} + 2\right)$$

Vậy quay $A(1,4)$ ngược chiều kim đồng hồ 1 góc 45° quanh $C(2,2)$, ta được điểm

$$B\left(-\frac{3\sqrt{2}}{2}+2, \frac{\sqrt{2}}{2}+2\right)$$



Nhấn vào **đây** để tương tác với hình trên Desmos.

```
Point Rotations(Point A, Point C, double rad) {
    Point A2 = A - C;
    Point B2 = Point(A2.x * cos(rad) - A2.y * sin(rad), A2.x * sin(rad) + A2.y
    Point B = B2 + C;
    return B;
}
```

Tổng hợp các link desmos trong bài

- Kiểm tra giao điểm của 2 đoạn thẳng (sử dụng tích có hướng)
- Đường tròn đi qua 3 điểm
- 2 đường thẳng vuông góc

- Phép đối xứng
- Phép quay (quanh tâm O)
- Phép quay (quanh điểm bất kì)

Luyện tập

Học phải đi đôi với hành, do đó mình đề xuất cho các bạn Codeforces Gym 100168 . Tuy đề bài trong gym được viết bằng tiếng Nga nhưng rất ngắn gọn và đi thẳng vào bài toán nên các bạn có thể dễ dàng google translate .

Bên dưới là một số bài tập có liên quan đến bài viết này, mình đã tóm tắt yêu cầu bài toán để các bạn có thể hiểu đề dễ dàng hơn.

- **Codeforces Gym - 100168K**: cho 2 đường thẳng có dạng $Ax + By + C = 0$, tìm giao điểm
- **CSES - Line Segment Intersection** : kiểm tra 2 đoạn thẳng có giao nhau hay không
- **Codeforces Gym - 100168M**: cho 2 điểm, viết phương trình đường thẳng
- **Codeforces Gym - 100168N**: cho 1 điểm và 1 vector pháp tuyến, viết phương trình đường thẳng
- **Codeforces Gym - 100168P**: kiểm tra điểm thuộc đường thẳng
- **Codeforces Gym - 100168Q**: kiểm tra điểm thuộc tia
- **Codeforces Gym - 100168R**: kiểm tra điểm thuộc đoạn
- **Codeforces Gym - 100168S**: kiểm tra 2 điểm có nằm cùng 1 phía với bờ là 1 đường thẳng có dạng $Ax + By + C = 0$
- **CSES - Point Location Test** : xác định vị trí của 1 điểm với 1 đường thẳng
- **CSES - Point in Polygon** : xác định vị trí của 1 điểm với 1 đa giác
- **Codeforces - Robo-Footballer** (vận dụng)