GeeksforGeeks
A computer science portal for geeks

GeeksQuiz

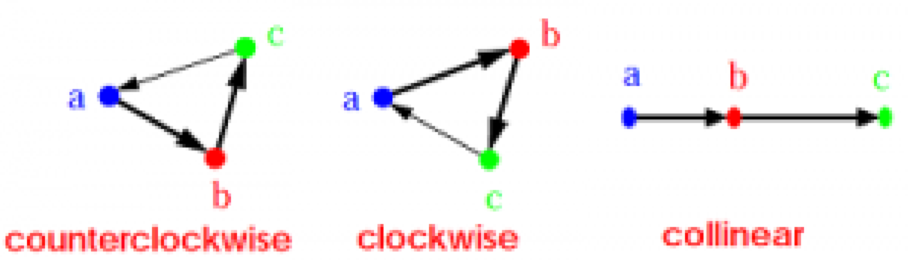
Login

Home Algorithms DS GATE Interview Corner Q&A C C++ Java Books Contribute Ask a Q About
Array Bit Magic C/C++ Articles GFacts Linked List MCQ Misc Output String Tree Graph

How to check if two given line segments intersect?

Given two line segments (p1, q1) and (p2, q2), find if the given line segments intersect with each other.

Before we discuss solution, let us define notion of **orientation**. Orientation of an ordered triplet of points in the plane can be
–counterclockwise
–clockwise
–colinear
The following diagram shows different possible orientations of (a, b, c)

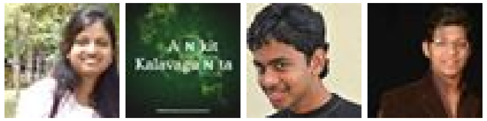


Note the word ‘ordered’ here. Orientation of (a, b, c) may be different from orientation of (c, b, a).

How is Orientation useful here?
Two segments (p1,q1) and (p2,q2) intersect if and only if one of the following two conditions is verified

1. *General Case:*
- (p1, q1, p2) and (p1, q1, q2) have different orientations and
- (p2, q2, p1) and (p2, q2, q1) have different orientations

Google™ Custom Search

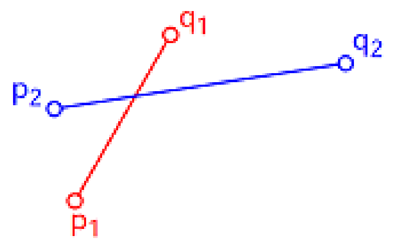


Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion
Geometric Algorithms

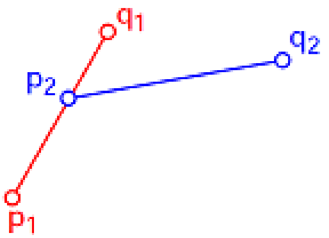
2. *Special Case*

- (p1, q1, p2), (p1, q1, q2), (p2, q2, p1), and (p2, q2, q1) are all collinear and
- the x-projections of (p1, q1) and (p2, q2) intersect
- the y-projections of (p1, q1) and (p2, q2) intersect

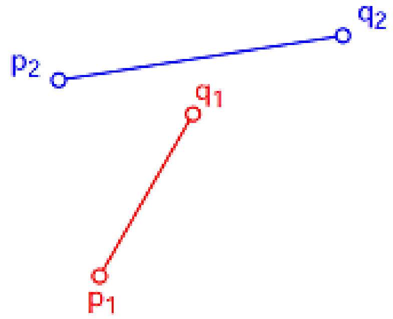
Examples of General Case:



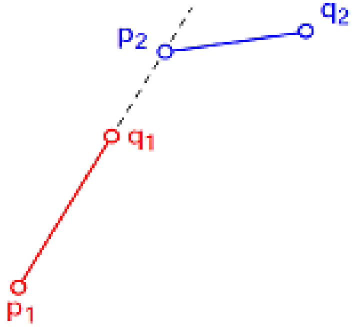
Example 1: Orientations of (p1, q1, p2) and (p1, q1, q2) are different. Orientations of (p2, q2, p1) and (p2, q2, q1) are also different



Example 2: Orientations of (p1, q1, p2) and (p1, q1, q2) are different. Orientations of (p2, q2, p1) and (p2, q2, q1) are also different

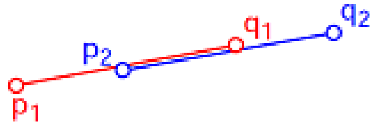


Example 3: Orientations of (p1, q1, p2) and (p1, q1, q2) are different. Orientations of (p2, q2, p1) and (p2, q2, q1) are same

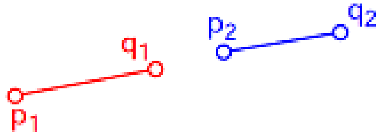


Example 4: Orientations of (p1, q1, p2) and (p1, q1, q2) are different. Orientations of (p2, q2, p1) and (p2, q2, q1) are same

Examples of Special Case:



Example 1: All points are colinear. The x-projections of (p1, q1) and (p2, q2) intersect. The y-projections of (p1, q1) and (p2, q2) intersect



Example 2: All points are colinear. The x-projections of (p1, q1) and (p2, q2) do not intersect. The y-projections of (p1, q1) and do not (p2, q2) intersect

Popular Posts

All permutations of a given string

Memory Layout of C Programs

Understanding “extern” keyword in C

Median of two sorted arrays

Tree traversal without recursion and without stack!

Structure Member Alignment, Padding and Data Packing

Intersection point of two Linked Lists

Lowest Common Ancestor in a BST.

Check if a binary tree is BST or not

Sorted Linked List to Balanced BST

Following is C++ implementation based on above idea.

```
// A C++ program to check if two given line segments intersect
#include <iostream>
using namespace std;

struct Point
{
    int x;
    int y;
};

// Given three colinear points p, q, r, the function checks if
// point q lies on line segment 'pr'
bool onSegment(Point p, Point q, Point r)
{
    if (q.x <= max(p.x, r.x) && q.x >= min(p.x, r.x) &&
        q.y <= max(p.y, r.y) && q.y >= min(p.y, r.y))
       return true;

    return false;
}

// To find orientation of ordered triplet (p, q, r).
// The function returns following values
// 0 --> p, q and r are colinear
// 1 --> Clockwise
// 2 --> Counterclockwise
int orientation(Point p, Point q, Point r)
{
    // See 10th slides from following link for derivation of the formu
    // http://www.dcs.gla.ac.uk/~pat/52233/slides/Geometry1x1.pdf
    int val = (q.y - p.y) * (r.x - q.x) -
              (q.x - p.x) * (r.y - q.y);

    if (val == 0) return 0;  // colinear

    return (val > 0)? 1: 2; // clock or counterclock wise
}

// The main function that returns true if line segment 'p1q1'
// and 'p2q2' intersect.
bool doIntersect(Point p1, Point q1, Point p2, Point q2)
{
    // Find the four orientations needed for general and
    // special cases
    int o1 = orientation(p1, q1, p2);
    int o2 = orientation(p1, q1, q2);
    int o3 = orientation(p2, q2, p1);
    int o4 = orientation(p2, q2, q1);

    // General case
    if (o1 != o2 && o3 != o4)
        return true;

    // Special Cases
    // p1, q1 and p2 are colinear and p2 lies on segment p1q1
    if (o1 == 0 && onSegment(p1, p2, q1)) return true;

    // p1, q1 and p2 are colinear and q2 lies on segment p1q1
    if (o2 == 0 && onSegment(p1, q2, q1)) return true;

    // p2, q2 and p1 are colinear and p1 lies on segment p2q2
    if (o3 == 0 && onSegment(p2, p1, q2)) return true;

     // p2, q2 and q1 are colinear and q1 lies on segment p2q2
    if (o4 == 0 && onSegment(p2, q1, q2)) return true;

    return false; // Doesn't fall in any of the above cases
}

// Driver program to test above functions
int main()
{
    struct Point p1 = {1, 1}, q1 = {10, 1};
    struct Point p2 = {1, 2}, q2 = {10, 2};

    doIntersect(p1, q1, p2, q2)? cout << "Yes\n": cout << "No\n";

    p1 = {10, 0}, q1 = {0, 10};
    p2 = {0, 0}, q2 = {10, 10};
    doIntersect(p1, q1, p2, q2)? cout << "Yes\n": cout << "No\n";

    p1 = {-5, -5}, q1 = {0, 0};
    p2 = {1, 1}, q2 = {10, 10};
    doIntersect(p1, q1, p2, q2)? cout << "Yes\n": cout << "No\n";

    return 0;
}
```

Output:

705 Subscribe

Recent Comments

Abhi You live US or India?

Google (Mountain View) interview · 9 minutes ago

Aman Hi, Why arent we checking for conditions...

Write a C program to Delete a Tree. · 49 minutes ago

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 52 minutes ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 1 hour ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

Count trailing zeroes in factorial of a number · 1 hour ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 1 hour ago

```
No
Yes
No
```

Sources:
http://www.dcs.gla.ac.uk/~pat/52233/slides/Geometry1x1.pdf
Introduction to Algorithms 3rd Edition by Clifford Stein, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

Related Tpoics:

- Backtracking | Set 8 (Solving Cryptarithmetic Puzzles)
- Tail Recursion
- Find if two rectangles overlap
- Analysis of Algorithm | Set 4 (Solving Recurrences)
- Print all possible paths from top left to bottom right of a mXn matrix
- Generate all unique partitions of an integer

- Russian Peasant Multiplication
- Closest Pair of Points | O(nlogn) Implementation

37 Tweet 3 5

Writing code in comment? Please use **ideone.com** and share the link here.

33 Comments **GeeksforGeeks**

Sort by Newest ▾

Join the discussion…

Kaidul Islam Sazal • 18 days ago

Is it possible in this code to determine in which point the lines intersect?

Amit • 6 months ago

onSsegment should be

if (p.x * (q.y - r.y) + q.x * (r.y - p.y) + r.x*(p.y - q.y) == 0)
return true;

Fredrik • 8 months ago

The code seems to be optimized for the case where the segments do interse
code for the case where the segments probably do not intersect?

clotho • 10 months ago

An easier way:
Segment L1 has edges A=(a1,a2), A'=(a1',a2').
Segment L2 has edges B=(b1,b2), B'=(b1',b2').

Segment L1 is the set of points tA'+(1-t)A, where 0<=t<=1.
Segment L2 is the set of points sB'+(1-s)B, where 0<=s<=1.
Segment L1 meet segment L2 if and only if for some t and s we have
tA'+(1-t)A=sB'+(1-s)B
The solution of this with respect to t and s is

t=((-b?'a?+b?'b?+b?a?+a?b?'-a?b?-b?b?')/(b?'a?'-b?'a?-b?a?'+b?a?-a?'b?'+a

s=((-a?b?+a?'b?-a?a?'+b?a?+a?'a?-b?a?')/(b?'a?'-b?'a?-b?a?'+b?a?-a?'b??+

So check if the above two numbers are both >=0 and <=1.
:)

Mayur Shah • 10 months ago

You can check it by using slope of lines whether the two lines are intersecting
slopes will never intersect at all and vice-versa!

GradLifeWoes ➦ Mayur Shah • 3 months ago

That would be the case if they are lines and not line segments.
Even in the case of lines another check has to be dome to determine t
So we should check they are not collinear and have same slopes to de

sarat G • 10 months ago

Hey,

What's the point in doing all this stuff...if the slopes of two line segments are e
any other case the two line segments always intersects..

```
/* Paste your code here (You may delete these lines if not writing co
```

kartik → sarat G · 10 months ago

That is why the topic says line *Segments* :)

4

Mohan · 10 months ago

I appreciate the efforts taken by geeksforgeeks community in posting useful pr

this one was very poorly explained.

??? · 10 months ago

the pdf is misleading, what he needs to say is : vector product.
check this wiki will explain everything:
http://en.wikipedia.org/wiki/C...

xxmajia · 10 months ago

For 10th page of http://www.dcs.gla.ac.uk/~pat/...

i don't get it, its true only when p, q, r 's x-projection are increasing

can anyone explain it a little bit?

```
/* Paste your code here (You may delete these lines if not writing co
```

kartik → xxmajia · 10 months ago

I think CLRS book would be helpful for better understanding.

IsAs · 10 months ago

distance(P1, Q1) = distance between P1 and Q1.
If point K lies on a line segment then following equation is true : distance(P1,Q

Find the slopes of two line segments and let's say if they are not equal - then

1) Form two equations of the form ax+by+c = 0
2) Compute the intersecting point by solving above line equations.Let's call ou
3) Verify whether distance(P1, Q1) == distance(P1,I) + distance(I,Q1)
4) Verify the same with P2 and Q2 as well
5) If intersecting point lies on both the line segments then above two verificatio

If slopes are equal then check if their equations are same or not. If equations a
no intersecting point

If equations are same then the points are collinear - verify the following (one of
lie on the other)

distance(P1,Q1) == distance(P1,P2) + distance(P2, Q1)
distance(P1,Q1) == distance(P1,Q2) + distance(Q2, Q1)

If any of the above statements are true then the line segments intersect

2

blackball • 10 months ago

How about this one:

```
 /* http://stackoverflow.com/questions/563198/how-do-you-detect-where

struct point {
    float x, y;
};

static inline int
```

```
line_intersect_2d(struct point a, struct point b,
                  struct point c, struct point d) {
    struct point cmp, r, s;
    cmp.x = c.x - a.x;
    cmp.y = c.y - a.y;
    r.x = b.x - a.x;
    r.y = b.y - a.y;
    s.x = d.x - c.x;
    s.y = d.y - c.y;
```

see more

Dixit Sethi • 10 months ago

Using basic maths, this can be done easily.

First take the two points of first line. say (x11, y11) and (x12, y12). This is line
of line passing through these points.

eqn: "y*(x12-x11)- x*(y12-y11) + x11(y12-y11) - y11(x12-x11)".

This eqn will equate to zero if x and y are points lying anywhere on the line (no
is above the line, then eqn value is positive and if the point is below the line, th

For cases like example 1, we need the two endpoints of the second line on the
vice versa. Check the value of the equation of first line for the two (x,y) end poi
values must be opposite in sign. Similarly check the two endpoints of the first
eqn values must also be opposite in sign. If thats the case, line segments are
case, stating it non intersecting.

Now, the case of both the endpoints of one line segment lying on the line of ot
example 1 and 2). This will cause all four equation values to be zero. This is h
all the points are collinear check is done for one of the endpoints lying between

see more

Parin · 10 months ago

How is orientation calculated?

1

GeeksforGeeks → Parin · 10 months ago

Please see slide 10 of the following link

http://www.dcs.gla.ac.uk/~pat/...

Parin → GeeksforGeeks · 10 months ago

Suppose the points are P1(-5,-5) P2(0,0) P3(-4,-3).
Then this is counterclockwise.But, here slope of p2p3 won't be

kartik → Parin · 10 months ago

The program seems to be printing counterclockwise on

Ravi Kesh Singh · 10 months ago

Another solution can be as following.

lets say

```
struct lineEqu{
int x;
int y;
int c; //line equ y-mx-c=0 here x =-m;y=1;z=-c.
}
```

populate lineEqu for(p1, q1) and (p2, q2) and write a method which return inte
(p1, q1) as well as (p2, q2) then they will intersect, if intersection point exist ar
then solution does not exist.

If intersection point does not exist then also sol does not exist.

Verma Shailendra · 10 months ago

agree it will always detect the intersection. but may give false result while not i

Hitesh · 10 months ago

If you want to apply your basic math skills, then solve an interesting problem c

In which you would find some interesting small problems like:

- Check whether the particular point lies inside the given polygon
- Find an angle between the 3 input points
- Given the input points, rearrange them in specific order to form the largest pc
- Solve the 2 lines for possible intersection

GeeksforGeeks → Hitesh · 10 months ago

@Hitesh: Thanks for sharing your inputs.
We have published an article on point and polygon problem.
We will soon be covering Convex Hull problem also.

You can also publish an article by mailing it to contribute@geeksforgee

Keep it up!

Uddhav · 10 months ago

I did not understand this condition " q.y = min(r.y, r.y) " ?
Why use max and min functions between same two values ??

Thank you

Hitesh · 10 months ago

```
class Point {

        private double x;
        private double y;

        public Point( ) {

        }

        public Point( double x, double y ) {

                this.x = x;
                this.y = y;

        }

        public double getX( ) {
```

see more

Sanjay Ahuja · 10 months ago

This approach will work but you may be missing one case when lines are para
projection may still intersect but lines are not intersecting.

Pratik · 10 months ago

Here orientation is nothing but the cross product.
Orientation (p,q,r) is cross product of Vector(p to q) and Vector(p to r).

But yeah. Good code!!

timus → Pratik · 10 months ago

THANKs, this really helped....

Abhilash Reddy · 10 months ago

in if statement you can simply put.
q.x == max(p.x, r.x) && q.y == max(r.y, r.y).

i don&#039t know about the implementation I was just curious to know code o
in more simplar way.

abhilash · 10 months ago

in frst if statement what is the point of keeping
q.x = min(p.x, r.x) &&
q.y = min(r.y, r.y)

you can simply keep q.x==max(p.x,r.x)&&q.y==min(r.y,r.y)

i dont know about the implementation of the program i was just curious to kno

Verma Shailendra · 10 months ago

I feel better and simpler solutions is we can find out only on the basis of projec

orientation at all).
ie (P1, Q1) and (P2, Q2) &#039s x projection (px1, qx1) will line on same side
not intersect, so if x and y both projection do not intersect than line segment d

Yelnil Gabo · 10 months ago

I like the colors. :P

cvr · 10 months ago

good code

Subscribe   Add Disqus to your site