

# GeeksforGeeks

A computer science portal for geeks

## GeeksQuiz

- [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](#)

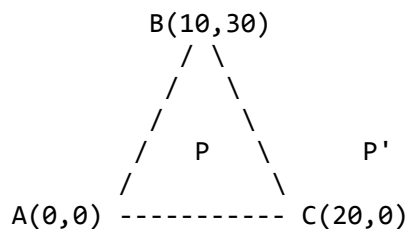
[Tree](#)

[Graph](#)

## Check whether a given point lies inside a triangle or not

Given three corner points of a triangle, and one more point P. Write a function to check whether P lies within the triangle or not.

For example, consider the following program, the function should return true for P(10, 15) and false for P'(30, 15)



Source: [Microsoft Interview Question](#)

### Solution:

Let the coordinates of three corners be  $(x_1, y_1)$ ,  $(x_2, y_2)$  and  $(x_3, y_3)$ . And coordinates of the given point P be  $(x, y)$

- 1) Calculate area of the given triangle, i.e., area of the triangle ABC in the above diagram. Area  $A = [x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2)]/2$
- 2) Calculate area of the triangle PAB. We can use the same formula for this. Let this area be  $A_1$ .
- 3) Calculate area of the triangle PBC. Let this area be  $A_2$ .
- 4) Calculate area of the triangle PAC. Let this area be  $A_3$ .
- 5) If P lies inside the triangle, then  $A_1 + A_2 + A_3$  must be equal to  $A$ .

```
#include <stdio.h>
#include <stdlib.h>

/* A utility function to calculate area of triangle formed by (x1, y1),
(x2, y2) and (x3, y3) */
float area(int x1, int y1, int x2, int y2, int x3, int y3)
{
    return abs((x1*(y2-y3) + x2*(y3-y1)+ x3*(y1-y2))/2.0);
}

/* A function to check whether point P(x, y) lies inside the triangle formed
by A(x1, y1), B(x2, y2) and C(x3, y3) */
bool isInside(int x1, int y1, int x2, int y2, int x3, int y3, int x, int y)
{
    /* Calculate area of triangle ABC */
    float A = area (x1, y1, x2, y2, x3, y3);

    /* Calculate area of triangle PBC */
    float A1 = area (x, y, x2, y2, x3, y3);

    /* Calculate area of triangle PAC */
    float A2 = area (x1, y1, x, y, x3, y3);

    /* Calculate area of triangle PAB */
    float A3 = area (x1, y1, x2, y2, x, y);

    /* Check if sum of A1, A2 and A3 is same as A */
    return (A == A1 + A2 + A3);
}

/* Driver program to test above function */
int main()
{
    /* Let us check whether the point P(10, 15) lies inside the triangle
    formed by A(0, 0), B(20, 0) and C(10, 30) */
    if (isInside(0, 0, 20, 0, 10, 30, 10, 15))
        printf ("Inside");
    else
        printf ("Not Inside");
}
```

```
    return 0;  
}
```

Ouptut:

Inside

**Exercise:** Given coordinates of four corners of a rectangle, and a point P. Write a function to check whether P lies inside the given rectangle or not.

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

## Related Topics:

- [Linearity of Expectation](#)
- [Iterative Tower of Hanoi](#)
- [Count possible ways to construct buildings](#)
- [Build Lowest Number by Removing n digits from a given number](#)
- [Set Cover Problem | Set 1 \(Greedy Approximate Algorithm\)](#)
- [Find number of days between two given dates](#)
- [How to print maximum number of A's using given four keys](#)
- [Write an iterative O\(Log y\) function for pow\(x, y\)](#)

Tags: [MathematicalAlgo](#)



1

Writing code in comment? Please use [ideone.com](#) and share the link here.

22 Comments

GeeksforGeeks

1 Login ▾

♥ Recommend 1

🔗 Share

Sort by Newest ▾



Join the discussion...



**Ram** • 4 months ago

we can easily solve it using Orientation. abc & abp; bca & bcp; bac & bap should have the same orientations.

^ | ▾ • Reply • Share ▸



**Shubham Goyal** • 8 months ago

Alternate : We cn also find out it by just making sure if none of (a,b,c,p) , (b,c,a,p) , (a,c,b,p) do intersect

^ | ▾ • Reply • Share ▸

**kkr** · a year ago

its giving "not inside" for the input {0,0 ,2 ,0,1,1} p{1,0} but as the point lie on the base of the triangle out put should be "inside"

^ | v · Reply · Share ›

**other** → kkr · 10 months ago

it will work...

```
#include<stdio.h>
```

```
int inside(int x1,int y1,int x2,int y2,int x3,int y3,int x,int y);
```

```
int area(int x1,int y1,int x2,int y2,int x3,int y3);
```

```
main()
```

```
{
```

```
if(inside(1,1,0,0,2,0,1,0))
```

```
{
```

```
printf("inside");
```

```
}
```

```
else
```

---

[see more](#)

3 ^ | v · Reply · Share ›

**Logic** · a year ago

I wrote a complete article about point in triangle test. It shows the barycentric, parametric and dot product based methods.

Then it deals with the accuracy problem occuring when a point lies exactly on one edge (with examples). Finally it exposes a complete new method based on point to edge distance.

<http://totologic.blogspot.fr/2...>

Enjoy !

1 ^ | v · Reply · Share ›

**mbrc** · 2 years ago

Will checking for equality of floating numbers work?

1 ^ | v · Reply · Share ›

**sachin** · 3 years ago



sachin · 2 years ago

Can anybody tell me if a point lies in a n vertices polygon?

How to solve?

^ | v · Reply · Share ›



mungruez → sachin · 2 years ago

I rewrote the code so now it can calculate if the point lies in a triangle whose sides can be very long much long.

```
#include<stdio.h>
#include <stdlib.h>
#include <inttypes.h>
/*NAME: Muhammad Zaakir Mungrue PROBLEM: 2153 -Playing with Triangles*/

long double area(int x1, int y1, int x2, int y2, int x3, int y3) {
    long double X= (x1*(y2-y3) + x2*(y3-y1)+ x3*(y1-y2))/2.0;
    if(X<0)return X*-1;
    return X;
}

int isInside(int x1, int y1, int x2, int y2, int x3, int y3, int x, int y) {
    long double A = area (x1, y1, x2, y2, x3, y3);
    long double A1 = area (x, y, x2, y2, x3, y3);
    long double A2 = area (x1, y1, x, y, x3, y3);
```

[see more](#)

^ | v · Reply · Share ›



abhishek08aug → sachin · 2 years ago

Will the same strategy not work?

1 ^ | v · Reply · Share ›



vishnu · 3 years ago

#include "stdafx.h"

int maxele=0;

int max(int a[])

{

for(int i=0;i&lt;a[i])

continue;

else

maxele=a[i];

}

return maxele;

}

```
int _tmain(int argc, _TCHAR* argv[])
{
    int a[3],b[3],resx,resy;
    for(int i=0;i<3;i++)
    {
```

---

[see more](#)[^](#) | [v](#) • [Reply](#) • [Share](#) ›**Vishnu** [→ vishnu](#) • 2 years ago

This wont work.

[^](#) | [v](#) • [Reply](#) • [Share](#) ›**duke** • 3 years ago

The above problem can also solve by taking the cross product of point to the one of the vertex of triangle and the two sides covering the point which gives the result that the point lies within the edges of triangle by doing the same process for three sides of triangle.

I am pasting my code here.....

```
#include
using namespace std;
typedef struct product
{
    int val[3];
};
product crossproduct(product x,product y)
{
    product z;
    z.val[0]=x.val[1]*y.val[2]-x.val[2]*y.val[1];
    z.val[1]=x.val[2]*y.val[0]-x.val[0]*y.val[2];
    z.val[2]=x.val[0]*y.val[1]-x.val[1]*y.val[2];
    return z;
}
```

---

[see more](#)[^](#) | [v](#) • [Reply](#) • [Share](#) ›**kartik** • 3 years ago

@Amandeep Singh &amp; @Hongliang

There should not be any precision problem because we divide integers by 2 and may get only 5 after after the point.

Also, I think use of floats can be avoided altogether by taking 2 times of areas. Please see the following program

following program.

```
#include <stdio.h>
#include <stdlib.h>

/* A utility function to calculate area of triangle formed by (x1, y1),
(x2, y2) and (x3, y3) */
int area(int x1, int y1, int x2, int y2, int x3, int y3)
{
    return abs((x1*(y2-y3) + x2*(y3-y1)+ x3*(y1-y2)));
}

/* A function to check whether point P(x, y) lies inside the triangle formed
```

[see more](#)

3 ^ | v • Reply • Share ›



**Amandeep Singh** • 3 years ago

We can actually make equations for three lines and substitute the value of P' in each equations. All the points lying on the same side of each line as that of P' will give same sign. this can be used effectively and will not involve float calculation. Hence no precision problem will occur.

^ | v • Reply • Share ›



**Hongliang** • 3 years ago

This solution is good, but it may lead to a precision problem: it introduces float number calculation, which may give, for example area = 100 and the summation of small areas = 99.999 or 100.001. It is hard to compare and may have potential pitfalls.

Actually in Quora someone summarized some beautiful answers to all possible cases:  
<http://www.quora.com/Algorithm...>

P.S. Your problem seems to put an triangle onto a lattice which has integer numbers for the positions. For this case, one can simply draw a vertical line and check how many (odd for outside, even for inside) intersections, it is a solution without float number calculation.

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

^ | v • Reply • Share ›



**Hongliang** ➔ Hongliang • 3 years ago

Sorry, I meant to: draw a vertical ray (not line) and check how many intersections. Odd for inside, even for outside. I was typing too fast, sorry.

^ | v • Reply • Share ›



**kafee** ➔ Hongliang • 3 years ago

I don't think odd or even will work

^ | v · Reply · Share ›



**rohan agrawal** · 3 years ago

@varun

If any one of  $A_1$ ,  $A_2$  or  $A_3 == 0$ , then the point is on the triangle.

^ | v · Reply · Share ›



**Pramod Ganapathi** · 3 years ago

A very good concept.

What about this idea?

A triangle is a region formed when three lines intersect each other. Each line is of the form  $f(x, y) = 0$ . When we plug in the new point  $(a, b)$  in  $f(x, y)$  we may get + or - depending on where the point is. We plug in the point in each of the three functions of the lines  $f_1(x, y)$ ,  $f_2(x, y)$  and  $f_3(x, y)$  and then check depending on signs whether the point is inside or outside.

I think the idea works but it should be formalized more rigorously.

^ | v · Reply · Share ›



**Varun** · 3 years ago

What about when point P is merged with any other point out of A,B, or C.

In this case P doesn't lies inside triangle ABC but still we will get output as true.

How this algorithm will take care of this case where P is at Boundary of ABC?

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

^ | v · Reply · Share ›



**rahul** → Varun · 3 years ago

Nim Ayyan

^ | v · Reply · Share ›



**Harish** → rahul · 3 years ago

just check if the area  $\neq 0$

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

^ | v · Reply · Share ›



Google™ Custom Search



- 
- 
- 
- - [Interview Experiences](#)
  - [Advanced Data Structures](#)
  - [Dynamic Programming](#)
  - [Greedy Algorithms](#)
  - [Backtracking](#)
  - [Pattern Searching](#)
  - [Divide & Conquer](#)
  - [Mathematical Algorithms](#)
  - [Recursion](#)
  - [Geometric Algorithms](#)
- 

## • 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](#)

•  Follow @GeeksforGeeks

## • Recent Comments

- radek

hey venki..is there a way to implement the same...

[The Ubiquitous Binary Search | Set 1](#) · [1 minute ago](#)

- radek

hey..is there a way to implement the same...

[The Ubiquitous Binary Search | Set 1](#) · [2 minutes ago](#)

- [Nikhil kumar](#)

public class...

[Print missing elements that lie in range 0 – 99](#) · [35 minutes ago](#)

- [Ashish Aggarwal](#)

Try Data Structures and Algorithms Made Easy -...

[Algorithms](#) · [57 minutes ago](#)

- Vlad

Thanks. Very interesting lectures.

[Expected Number of Trials until Success](#) · [2 hours ago](#)

- [cfh](#)

My implementation which prints the index of the...

[Longest Even Length Substring such that Sum of First and Second Half is same](#) · [2 hours ago](#)

•

@geeksforgeeks, [Some rights reserved](#) [Contact Us!](#)

Powered by [WordPress](#) & [MooTools](#), customized by geeksforgeeks team