# Clipping Line on Circle

If both the starting point and ending point of the line are outside the circle the no clipping is necessary.

Otherwise we will run intersection to find the points.

* No points or one point -> no clipping
* Two points. Only care about the points that are on the actual segment.

## Point inside circle test:

To test if a point is inside the circle, we use the equation of the circle.

Where is the origin of the circle, is the radius.

For simplicity, we will consider the origin of the circle as and the line coordinates as relative to the original origin.

Simply plug in the coordinates and return the inequality result.

## Intersection between line and circle

Circle equation:

Line equation: .

To find the values of the constants, we will use the original line equation using the two points defining the line:

So , and

To find the intersection point, we will solve the two equations together.

As a reminder, the solutions for a general equation of the form

If we plug in the expressions into the above formula we get:

We will assign .

Substituting in the line equation we get:

Notice that the line doesn’t intersect the circle if

## Point on segment

If a point is on a line, to check if it is on a particular segment spanning to . We simply check if the point is “between” start and finish. To do it programmatically, we assert that the start is closer to the top left corner and compare the points as a pair.

|  |
| --- |
| bool pointOnSegment(const point& r, point st, point en) {  if (st > en) swap(st, en);  return r >= st && r >= en;  } |

# Clipping Circle on Circle

\*no idea\*

# Clipping Bezier on Circle

\*no idea\*