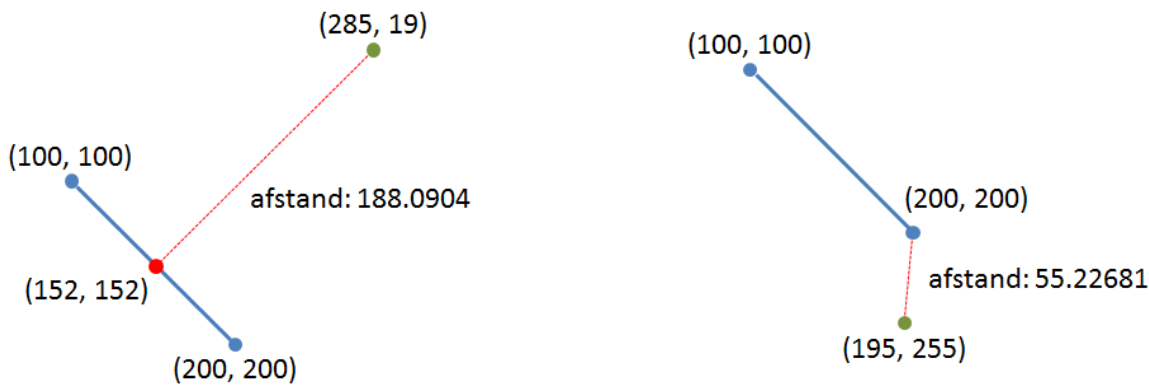


# Shortest distance to line segment

What is the shortest distance from a point to a line segment? On determining this distance, two scenarios should be distinguished, as given in the picture below.



Assume that the line segment joins the points  $(x_1, y_1)$  and  $(x_2, y_2)$ , and that we are searching for the shortest distance between this line segment and a point  $(x_3, y_3)$ . In the first scenario (left picture), the perpendicular line from the point  $(x_3, y_3)$  intersects the straight line on which the line segment is situated in point  $(x_v, y_v)$  on the line segment. This point is called the **nadir**. The shortest distance then equals the distance between the points  $(x_3, y_3)$  and  $(x_v, y_v)$ . In the other scenario (right picture), the nadir is located outside the line segment, and the shortest distance is formed by the shortest distance from point  $(x_3, y_3)$  to each end of the line segment  $(x_1, y_1)$  and  $(x_2, y_2)$ .

## Assignment

- Write a function `distance` to which four integers should be given. These numbers respectively represent the co-ordinates of two points  $(x_1, y_1)$  and  $(x_2, y_2)$ . The function should return the Euclidean distance  $d$  between those points, that is given by the formula  $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$
- Write a function `nadir` to which six integers should be given. These numbers respectively represent the coordinates of three points  $(x_1, y_1)$ ,  $(x_2, y_2)$  and  $(x_3, y_3)$ . The function should return two (real) co-ordinates  $(x_v, y_v)$  of the nadir. The nadir is the intersection of the perpendicular from the point  $(x_3, y_3)$  with the straight line through the points  $(x_1, y_1)$  and  $(x_2, y_2)$ , and the line segment that joins the points  $(x_1, y_1)$  and  $(x_2, y_2)$ . In order to do so, we first calculate the value  $u = \frac{(x_3 - x_1)(x_2 - x_1) + (y_3 - y_1)(y_2 - y_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ . The perpendicular line will only intersect the line segment if  $0 \leq u \leq 1$ . If that isn't the case, the function should return the value `None` twice for the coordinates of the nadir. If both ends of the line segment coincide, the denominator in the calculation of  $u$ , equals zero, but the nadir also coincides with the ends of the line segment. In other cases, the co-ordinates of the nadir are given by 
$$\begin{array}{l} x_v = x_1 + u(x_2 - x_1) \\ y_v = y_1 + u(y_2 - y_1) \end{array}$$
- Use the functions `distance` and `nadir` to write a function `shortest_distance` to which six integers should be given. These numbers respectively represent the co-ordinates of three points  $(x_1, y_1)$ ,  $(x_2, y_2)$  en  $(x_3, y_3)$ . The function should return the shortest distance between point  $(x_3, y_3)$  and the line segment that joins the points  $(x_1, y_1)$  and

$(x_2, y_2)$ .

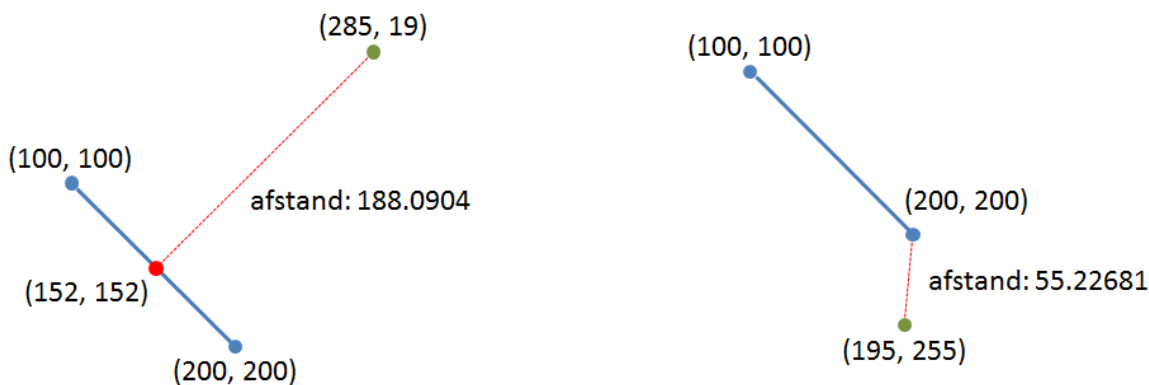
## Example

```
>>> distance(152, 152, 285, 19)
188.09040379562165
>>> distance(100, 100, 195, 255)
181.79658962697843
>>> distance(200, 200, 195, 255)
55.226805085936306
```

```
>>> nadir(100, 100, 200, 200, 285, 19)
(152.0, 152.0)
>>> nadir(100, 100, 200, 200, 195, 255)
(None, None)
```

```
>>> shortest_distance(100, 100, 200, 200, 285, 19)
188.09040379562165
>>> shortest_distance(100, 100, 200, 200, 195, 255)
55.226805085936306
```

Wat is de kortste afstand van een punt tot een lijnsegment? Bij het bepalen van deze afstand moeten er twee scenario's onderscheiden worden, zoals aangegeven in onderstaande figuur.



Stel dat het lijnsegment de punten  $(x_1, y_1)$  en  $(x_2, y_2)$  met elkaar verbindt, en dat we de kortste afstand zoeken tussen dit lijnsegment en het punt  $(x_3, y_3)$ . In het eerste scenario (figuur links) snijdt de loodlijn uit het punt  $(x_3, y_3)$  de rechte waarop het lijnsegment gelegen is in een punt  $(x_v, y_v)$  dat zich op het lijnsegment bevindt. Dit punt wordt het **voetpunt** genoemd. De kortste afstand is dan gelijk aan de afstand tussen de punten  $(x_3, y_3)$  en  $(x_v, y_v)$ . In het tweede scenario (figuur rechts) ligt het voetpunt buiten het lijnsegment, en wordt de kortste afstand gevormd door de kleinste afstand van het punt  $(x_3, y_3)$  tot elk van de twee uiteinden van het lijnsegment  $(x_1, y_1)$  en  $(x_2, y_2)$ .

## Opgave

- Schrijf een functie `afstand` waaraan vier natuurlijke getallen moeten doorgegeven worden. Deze getallen stellen respectievelijk de coördinaten van twee punten  $(x_1, y_1)$  en  $(x_2, y_2)$  voor. De functie moet de Euclidische afstand  $d$  tussen deze twee punten teruggeven, die gegeven wordt door de formule  $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$
- Schrijf een functie `voetpunt` waaraan zes natuurlijke getallen moeten doorgegeven worden. Deze getallen stellen respectievelijk de coördinaten van drie punten  $(x_1, y_1)$ ,  $(x_2,$

$y_2$ ) en  $(x_3, y_3)$  voor. De functie moet de twee (reële) coördinaten  $(x_v, y_v)$  van het voetpunt teruggeven. Dit voetpunt is het snijpunt van de loodlijn uit het punt  $(x_3, y_3)$  met de rechte door de punten  $(x_1, y_1)$  en  $(x_2, y_2)$ , en het lijnsegment dat de punten  $(x_1, y_1)$  en  $(x_2, y_2)$  verbindt. Hiervoor berekent men eerst de waarde  $u = \frac{(x_3 - x_1)(x_2 - x_1) + (y_3 - y_1)(y_2 - y_1)}{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ . De loodlijn zal enkel het lijnsegment snijden als  $0 \leq u \leq 1$ . Indien dat niet het geval is, dan moet de functie tweemaal de waarde `None` teruggeven voor de coördinaten van het voetpunt. Als de twee uiteinden van het lijnsegment samenvallen, dan wordt de noemer bij de berekening van  $u$  gelijk aan nul, maar dan valt ook het voetpunt samen met de uiteinden van het lijnsegment. In de andere gevallen worden de coördinaten van het voetpunt gegeven door 
$$\begin{eqnarray*} x_v &=& x_1 + u(x_2 - x_1) \\ y_v &=& y_1 + u(y_2 - y_1) \end{eqnarray*}$$

- Gebruik de functies `afstand` en `voetpunt` om een functie `kortste_afstand` te schrijven waaraan zes natuurlijke getallen moeten doorgegeven worden. Deze getallen stellen respectievelijk de coördinaten van drie punten  $(x_1, y_1)$ ,  $(x_2, y_2)$  en  $(x_3, y_3)$  voor. De functie moet de kortste afstand teruggeven tussen het punt  $(x_3, y_3)$  en het lijnsegment dat de punten  $(x_1, y_1)$  en  $(x_2, y_2)$  verbindt.

## Voorbeeld

```
>>> afstand(152, 152, 285, 19)
188.09040379562165
```

```
>>> afstand(100, 100, 195, 255)
181.79658962697843
```

```
>>> afstand(200, 200, 195, 255)
55.226805085936306
```

```
>>> voetpunt(100, 100, 200, 200, 285, 19)
(152.0, 152.0)
```

```
>>> voetpunt(100, 100, 200, 200, 195, 255)
(None, None)
```

```
>>> kortste_afstand(100, 100, 200, 200, 285, 19)
188.09040379562165
```

```
>>> kortste_afstand(100, 100, 200, 200, 195, 255)
55.226805085936306
```